

Rhein Tech Laboratories, Inc.  
360 Herndon Parkway  
Suite 1400  
Herndon, VA 20170  
<http://www.rheintech.com>

Client: Full Spectrum Inc.  
Models: BS100 & MS400  
FCC ID: X27-FS-NPCS000  
Standard: FCC Part 24  
Report #: 2009306

## **Appendix J : Manual**

Please refer to the following pages.

# FullMax BS100 and MS400 User Manual

Version 1.1

March 9, 2010

**Table of Contents**

1 Overview.....3

2 FullMax Description .....4

**2.1 System Description .....4**

**2.2 BS100 and MS400 Hardware Architecture .....5**

**2.3 FullMax BS100 and BS400 Software Architecture..... 11**

3 FullMax BS100 and MS400 Embedded Software Functionality..... 12

4 FullMax System Installation and Commissioning..... 14

**4.1 Enclosure of BS100 and MS400 ..... 14**

**4.2 Connecting the BS100 and MS400..... 14**

**4.3 FullMax BS100 & MS400 Tune Up and Alignment Procedures..... 15**

**4.4 CLI Commands for BS100 ..... 16**

**4.5 Monitoring the BS / MS with FullMax NMS ..... 17**

5 FullMax system operation..... 18

**5.1 Operation Restrictions ..... 18**

**5.2 BS100 Operation..... 19**

    5.2.1 Main Group..... 19

    5.2.2 mac-config ..... 21

    5.2.3 ul-config..... 28

    5.2.4 dl-config..... 40

    5.2.5 zone-config ..... 45

    5.2.6 bs-cap-config..... 51

    5.2.7 bs-burst-profile..... 53

    5.2.8 bs-ss-action ..... 56

    5.2.9 ss-registered ..... 57

    5.2.10 ss-ip..... 59

    5.2.11 device ..... 61

    5.2.12 Measurements ..... 64

    5.2.13 pkm-config ..... 67

    5.2.14 bs-private..... 70

    5.2.15 trap-config..... 76

    5.2.16 lapc-config ..... 78

**5.3 MS400 Operation ..... 92**

    5.3.1 Main Group..... 92

    5.3.2 ss-config..... 93

    5.3.3 device ..... 99

    5.3.4 Measurements ..... 102

    5.3.5 ss-private..... 105

    5.3.6 ss-chconfig ..... 111

    5.3.7 ss-trap..... 114

6 Annex A: Frequency Reuse Schemes ..... **Error! Bookmark not defined.**

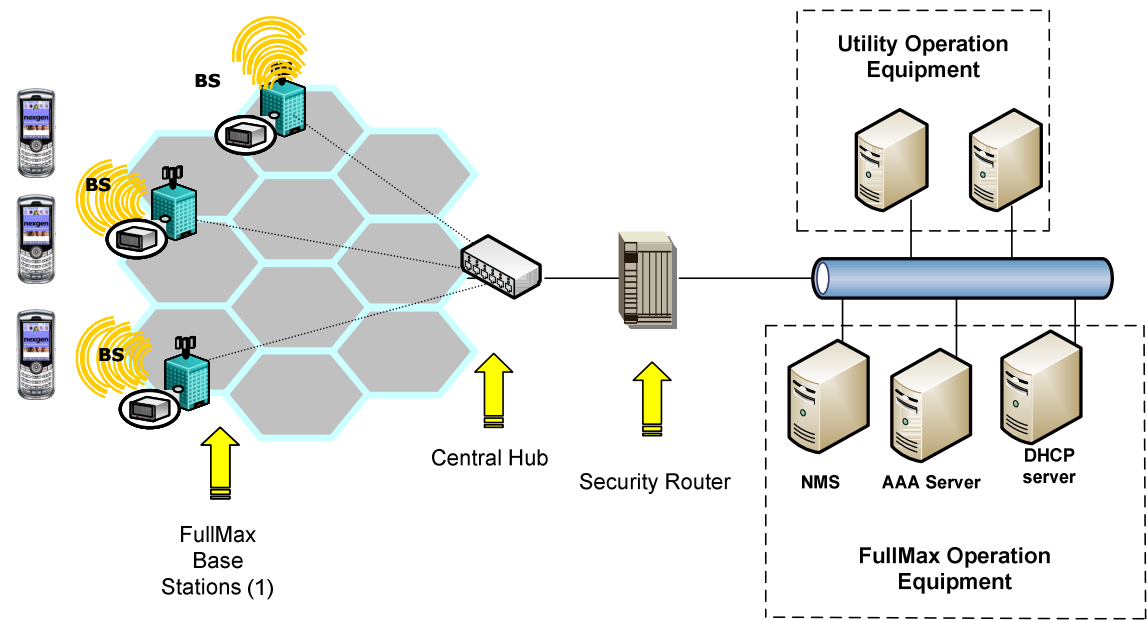
# 1 Overview

- a) FullMax is a Point to Multipoint (PtMP) broadband wireless system based on the WiMAX-e (IEEE 802.16e-2005) protocol with modifications needed to enable its operation in the Narrow PCS band (NPCS). The system consists of BS100 Base Stations, MS400 Mobile or Fixed subscriber stations, standard backhaul networking equipment connecting the BS100 Base Stations to a Router at the Network Operating Center (NOC) and a Network Management System (NMS).
- b) FullMax main modifications relative to the IEEE802.16e-2005 air interface protocol are:
  - 1) The RF front end can tune to the enter frequencies within the following NPCS channels:
    - a. 940.3000 – 940.8000 (500 kHz)
    - b. 901.3000 – 901.7500 (450 kHz)
    - c. 901.3000 – 901.7000 (400 kHz)
    - d. 901.3500 – 901.7500 (400 kHz)
    - e. 930.0000 – 930.4000 (400 kHz)
    - f. 930.6000 – 931.0000 (400 kHz)
    - g. 940.3000 – 940.7500 (450 kHz)
    - h. 940.3000 – 940.7000 (400 kHz)
    - i. 940.3500 – 940.7500 (400 kHz)
  - 2) The BS100& MS400 operate in 400 KHz, 450 KHz and 500 KHz wide channels which are available in the NPCS band.
  - 3) The BS100 and MS400 have analog and digital filters which are designed to meet the FCC Part 24 spectral mask in the transmit direction and the channel selectivity in the receive direction.
- c) Outline of the document:
  - 1) Paragraph 2: FullMax system description
  - 2) Paragraph 3: FullMax system operation
  - 3) Paragraph 4: BS100 and MS400 installation and commissioning.

## 2 FullMax Description

### 2.1 System Description

- a) FullMax system architecture is described in figure 2-1 below. The system is used to establish private, multi-cell, Point to Multipoint broadband wireless service for electrical utilities and other mission critical industries. It supports both fixed and mobile applications. FullMax BS's are typically installed in the existing Private Land Mobile Radio (PLMR) towers serving their respective cells. Wireless backhaul facilities are used to connect the BS's to the central site of the system. FullMax fixed and mobile SS's are deployed throughout the tower's serving area.



(1) FullMax base station consists of 3 independent base station sectors and a base station hub

Fig. 2-1: FullMax system architecture

- b) FullMax operates in unpaired spectrum using Time Division Duplex (TDD). FullMax also operates in paired spectrum employing each portion of the paired spectrum as an independent unpaired spectrum.
- c) FullMax BS100 is designed to support a single sector and as such, it supports the construction of a Base Station with multiple sectors (1, 2, 3 or more). An external bridge/router is used to forward/route the traffic to the appropriate sector. Note

that the most common deployment is a 3 sector configuration. The sector configuration dictates the type of antenna that should be used.

- d) FullMax supports various frequency reuse schemes as outlined in Annex A to this document.
- e) The FullMax BS antenna is typically installed at an existing utility tower. The antenna should be installed as high on the tower as possible. The BS also employs a GPS antenna for TDD framing synchronization.
- f) FullMax MS400 can be used as either a fixed indoor Subscriber Station with an outdoor antenna or as a Mobile Station in a Utility truck. As a fixed SS, MS400 supports both Ethernet 100 Base T connectivity as well as serial RS232 connectivity. The serial connectivity is needed to support various legacy applications like Supervisory Control And Data Acquisition (SCADA) applications.

## **2.2 BS100 and MS400 Hardware Architecture**

- a) The FullMax BS100 & MS400 radio architecture is described in figure 2-1 and 2-2 below. It consists of a Baseband Processor Board (BBP), an Analog Front End (AFE) section and a Low Voltage Power Supply (LVPS) board.
- b) The BBP block diagram is described in figure 2-3. It is the heart of the FullMax radio. It is designed to perform MAC, PHY, networking, network management and other functions that are required in a broadband wireless BS and MS. The BBP has the following main characteristics:
  - 1) Processing resources:
    - a. A TI DSP and a Xilinx Spartan 3A FPGA to execute the PHY layer
    - b. A Freescale PQ3 processor to execute the MAC layer and complementary embedded software
  - 2) A GPS time reference module is available for synchronization<sup>1</sup> and for location based services.
  - 3) User interfaces: 100 Base T, RS232
  - 4) Interface to the AFE is done through a digital I/Q interface.
- c) The AFE section block diagram is described in figure 2-4. The AFE section performs signal processing functions that are needed to deliver the signal to the antenna and to receive the signal from the antenna. The AFE consists of:
  - 1) A RF Small Signal (RFSS) board which contains a baseband section, an IF section and an RF section.

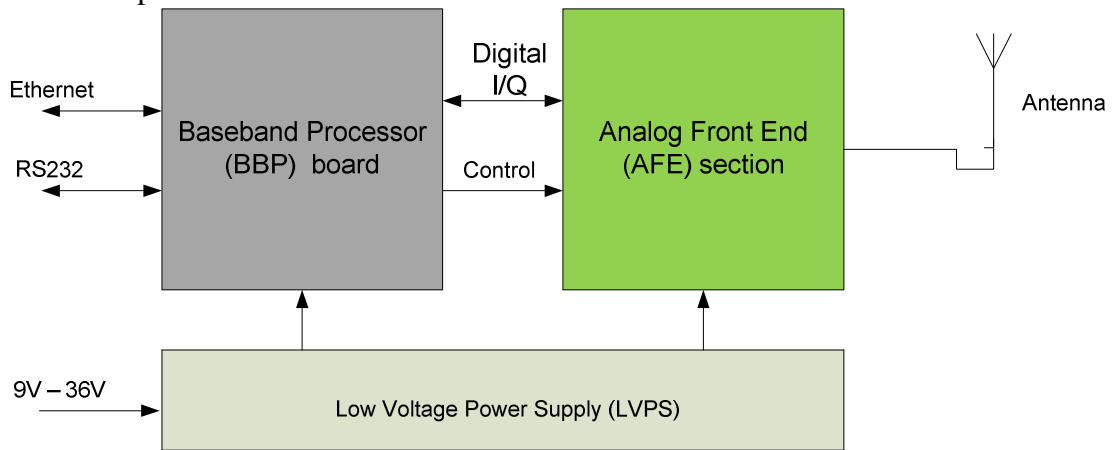
---

<sup>1</sup> e.g. for TDD frame synchronization.

- a. The baseband section consists of an A/D, D/A, a programmable receive baseband filter, a Digital Pre Distortion (DPD) Equalizer a programmable sampling clock synthesizer and a FPGA.
  - b. The IF section consists of an IF LO, an I/Q modulator/demodulator, an IF receive channel bank and an IF transmit filter.
  - c. The RF section consists of a RF LO, an RF up/down convertor, an RF receive channel bank and a RF transmit filter.
- 2) A RF Front End (RFFE) board which contains a RF PA, LNA, AGC and T/R switch.
  - 3) The AFE employs three 8051 microcontrollers for monitoring and control of all aspects of the AFE operation. A serial interface protocol is available to support control of the master housekeeping microcontroller on the RFSS board by the main PQ3 processor on the BBP board.

**Note: The AFE employs a non agile external RF bandpass filter which is shown as a yellow block in the AFE block diagram in figure 4.**

- d) The Low Voltage Power Supply (LVPS) block diagram is described in figure 2-5. The LVPS generates all the voltages that are needed at the BBP and the AFE. It is designed for an input voltage range of 9 to 36 VDC to support power feed for a vehicle battery. An optional external power brick is available if needed to support AC or -48 VDC power feed.



**Figure 2-1: FullMax BS100 & MS400 High Level Architecture**

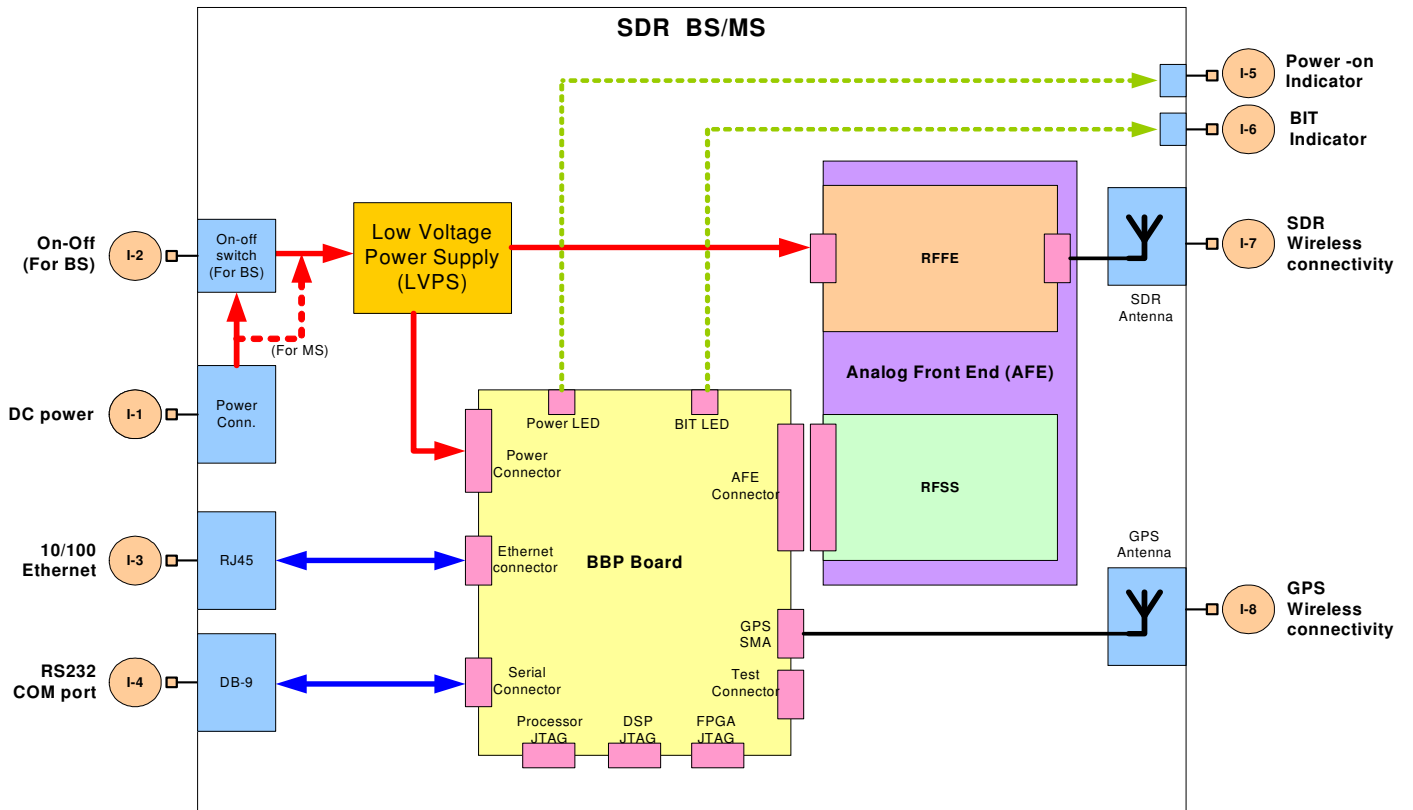


Figure 2-2: FullMax BS100 & MS400 Block Diagram



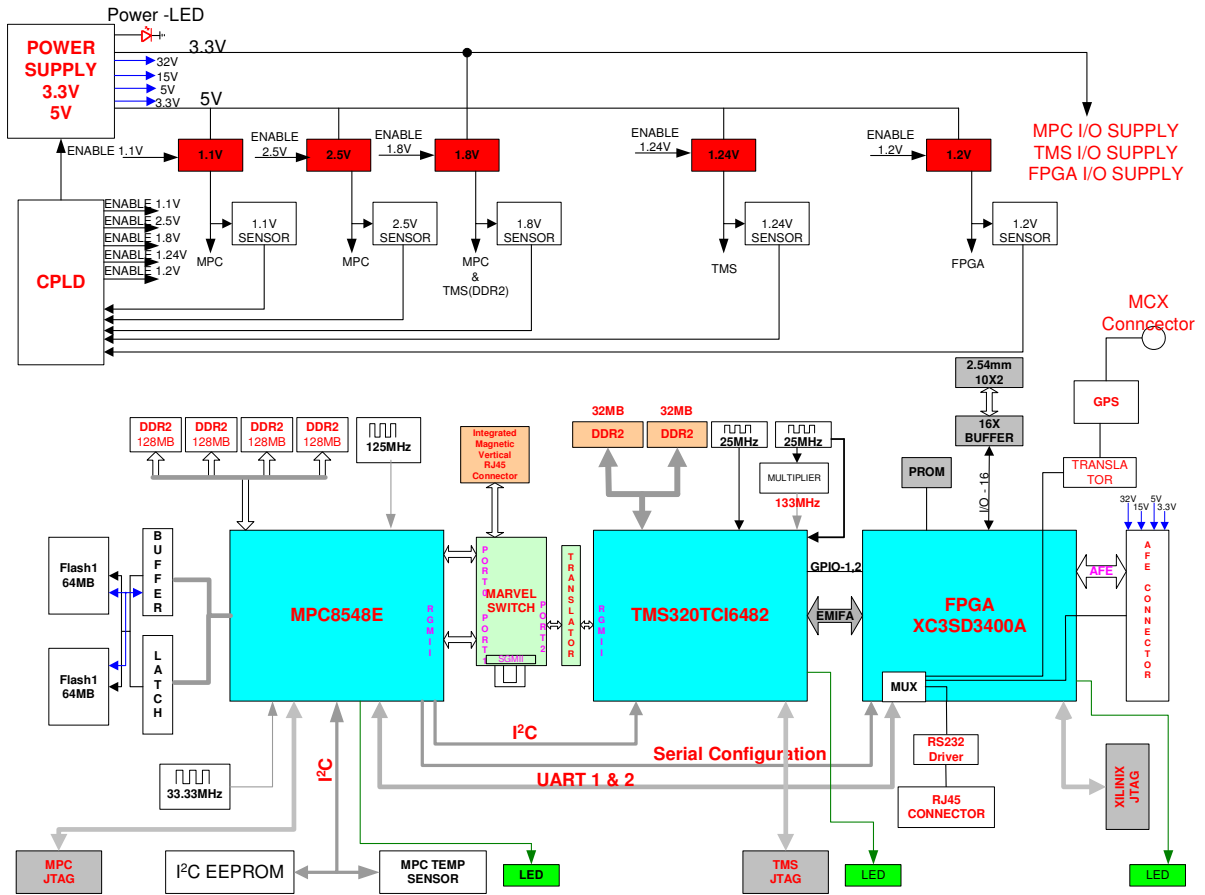


Figure 2-3: Baseband Processor (BBP) Board Block Diagram

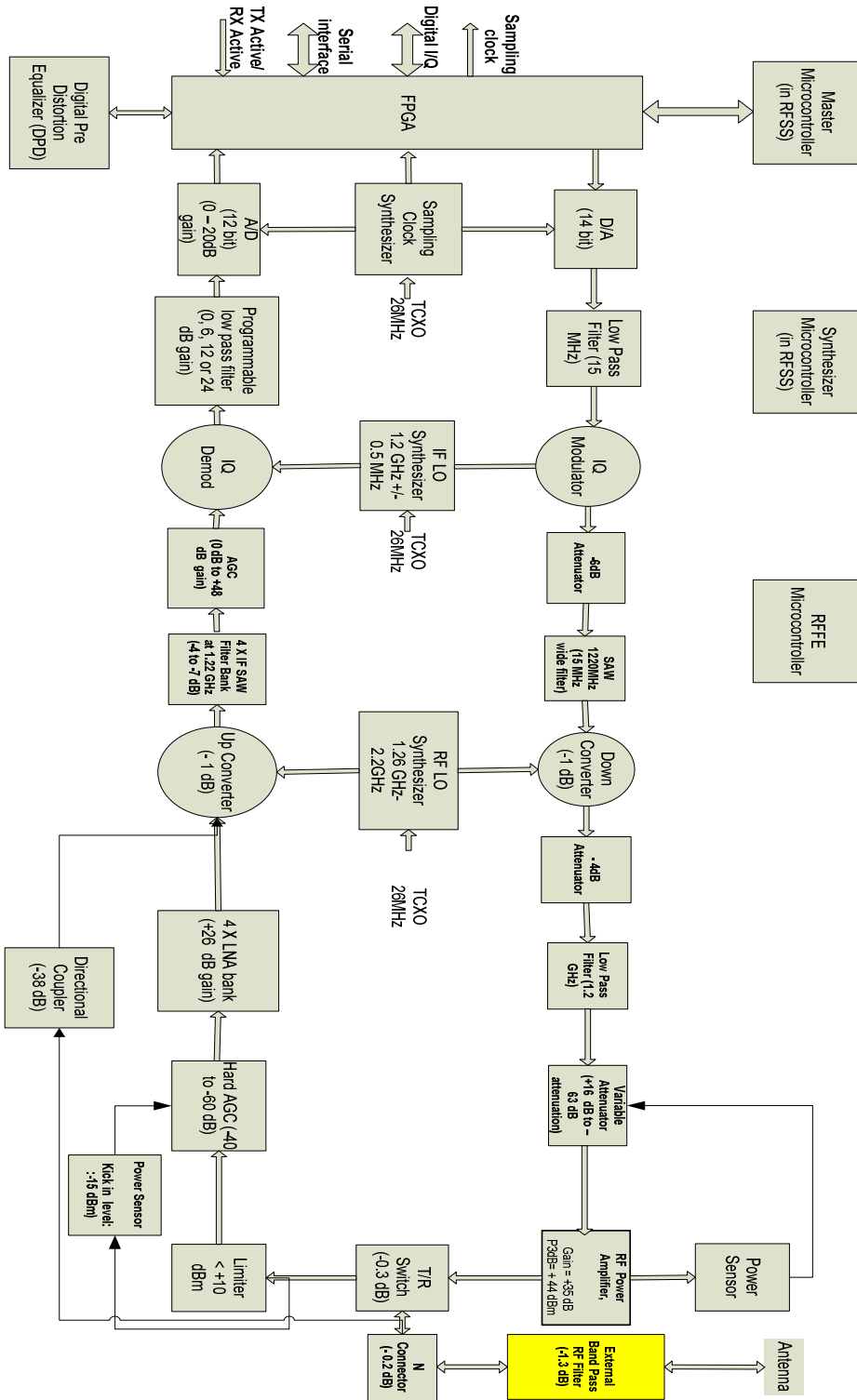


Figure 2- 4: Analog Front End (AFE) Block Diagram

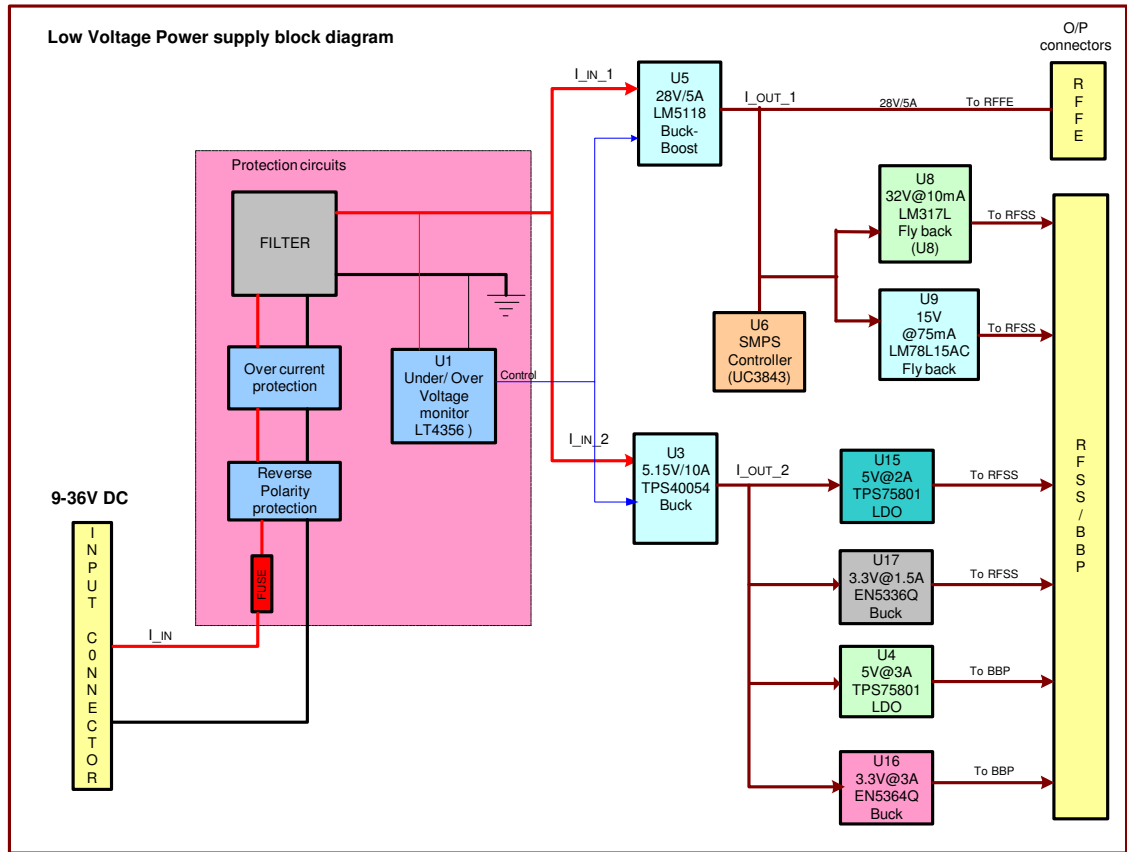


Figure 2-5: Low Voltage Power Supply (LVPS) Block Diagram

### 2.3 FullMax BS100 and BS400 Software Architecture

- a) FullMax BS100 and MS400 software architecture is described in figure 2-6 below. It consists of PHY layer, MAC layer and general purpose embedded software components.

Modified WiMAX-e PHY layer SW	Modified WiMAX-e MAC Layer SW	General purpose embedded software
Basic Software Tools	Monta Vista Linux OS & BSP	
TI 6482 Himalaya DSP & Xilinx Spartan 3A FPGA	PowerQuicc III 8548E processor	

**Figure 2-6: FullMax BS100 & MS400 software architecture**

### 3 FullMax BS100 and MS400 Embedded Software Functionality

- a) FullMax BS100 and MS400 support the MAC and PHY layer requirements as defined in the “WiMAX Forum Mobile System Profile Release 1.0 Approved Specifications”. This section describes the complementary embedded software that is not considered part of the MAC and the PHY layer software.
- b) Networking:**
- Supports communication over a 100 Base T Ethernet interface
  - Supports IP host functionality: BS100 & MS400 have an IP address and they support the following IP protocols: DHCP, ARP, ICMP, TFTP, FTP, SNMP, ToD (RFC-868). Also, BS100 and MS400 have a read only MAC address which is programmable during manufacturing.
  - Learning bridge and layer 2 forwarding: BS100 and MS400 support learning bridge functionality, maintain a Table of Connected Entities (TCE) and use it to forward packets to the correct destination (Ethernet interface, wireless interface or IP host)
- c) QOS:**
- Most of the QOS functionality is considered part of the MAC layer. The general purpose embedded software provides complimentary QOS functionality such as traffic prioritization.
- d) Security:**
- The general embedded software supports encryption and authentication functionality and processes which are complimentary to the MAC layer security functionality.
- e) Provisioning:**
- IP address provisioning (the BS100 and MS400 acquire an IP address from an external standard DHCP server).
  - Time provisioning (the BS100 and MS400 acquire date and time from an RFC-868 Time server).
  - EAP Provisioning (The BS100 and MS400 acquire the MSK Key from the AAA server).
  - Configuration provisioning (The BS100 and MS400 configuration provisioning is done through CLI and through SNMP commands).
  - QoS Provisioning (The BS100 and MS400 QOS provisioning is done through SNMP commands).

**f) Channel Acquisition**

- The MS400 supports a pre-configured channel acquisition plan, i.e., a preconfigured list of channel alternatives, characterized by their center frequency and the bandwidth. The MS400 goes through the list and performs successive channel acquisition attempts until an attempt is successful.
- Criteria for channel acquisition success:
  - Successful registration if the MS400 was not registered prior to channel acquisition.
  - Successful ranging if the MS400 was registered prior to channel acquisition (i.e., in the case of moving to a new BS100).
- Once a channel is successfully acquired, the parameters of the acquired channel are saved to the MS400 flash memory and are used as first acquisition option on MS400 power on.

**g) Network Management Support**

- The BS100 and MS400 have a SNMPv2c agent and they can be remotely managed by FullMax Network Management System outlined in Paragraph 4 below.
- The BS100 and MS400 also support a Telnet based Command Line Interface (CLI) protocol which can be used to configure all BS100 and MS400 parameters.

**h) Miscellaneous BS100 and MS400 hardware platform Management**

- Remote and secure software download
- Initialization and configuration Persistence: The BS100 and MS400 support a CLI command that initiates the saving of the configuration in the flash. Saved configuration is used after power off.

## 4 FullMax System Installation and Commissioning

### 4.1 Enclosure of BS100 and MS400

The BS100 and MS400 devices share the same enclosures



Figure 1: Top View



Figure 2: Bottom View (with FCC label)



Figure 3: Front View



Figure 4: Rear View



Figure 5: Right Side View



Figure 6: Left Side View

### 4.2 Connecting the BS100 and MS400

1. Place the enclosure in a stable location. Make sure that air can run freely to the fans on the rear of the enclosure and the holes on the enclosures sides.
2. Connect the cable from the Antenna to the device RF connector on the enclosure left side.

3. Use an Ethernet cable to connect the device with end equipment
4. Connect the power cable to the power connector on the enclosure left side.
5. Connect the power label to the power source.

### **4.3 FullMax BS100 & MS400 Tune Up and Alignment Procedures**

1. Frequency and transmit power configuration of the FullMax BS100 and MS400 is done by means of either a Command Line Interface (CLI) tool or a Web based Network Management (NMS) tool.
2. The NMS/CLI tools is used to configure the following parameters:
  - a. BS100 parameters:
    - i. Center RF frequency
    - ii. Channel bandwidth
    - iii. Transmit power
  - b. MS400 parameters:
    - i. Maximum uplink transmit power
    - ii. An automatic channel acquisition table with up to 10 entries. Each entry includes the center frequency and the channel bandwidth.
3. The center frequency, channel bandwidth and transmit power at the BS100 are configured statically by the NMS/CLI tool. This means that the values will be maintained unless they are changed by the CLI/NMS tool.
4. The MS400 on the other hand, performs automatic channel acquisition by selecting dynamically, the best channel (highest RSSI) from a list of up to 10 pre-configured entries (see paragraph 2bi above). The transmit power of the MS is controlled by an automatic closed loop power control algorithm running at the BS100. The transmit power of the MS will be increased by the BS100 as the distance to the MS is increased (i.e., as the path loss increases and the RSSI as measured by the BS100 becomes lower). Note however that the MS will not transmit above the maximum uplink transmit power (see paragraph 2bii) even if ordered by the BS100.
5. Both BS and MS employ an external RF bandpass filter as described in Annex A to this document. This is an additional safeguard which guarantees that the BS100 and the MS400 will not transmit outside the band.



6. The remainder of the document describes the CLI commands that are used to configure the BS100 and MS400 parameters described above.

#### **4.4 CLI Commands for BS100**

After logging in through telnet

*[FULLMAX]\$* Prompt for user is displayed.

Type help or ? to look at different groups.

Lock to dl-config group using dl-config command

*[FULLMAX]\$*

*[FULLMAX]\$ dl-config*

*You are locked to dl-config group.  
Only dl-config operations are allowed.  
Use help or ? for help.*

*[FULLMAX(dl-config)]\$*

##### **For Frequency:**

*[FULLMAX(dl-config)]\$ show center-freq  
center-freq 940500 khz*

*[FULLMAX(dl-config)]\$ set center-freq  
updated center-freq 930800 khz*

##### **For Tx Power :**

*[FULLMAX(dl-config)]\$ show tx\_power  
tx\_power 9 dbm*

*[FULLMAX(dl-config)]\$ set tx\_power 16  
updated tx\_power 16 dbm*

##### **For Channel Bandwidth:**

*[FULLMAX(dl-config)]\$ show bandwidth  
bandwidth 500 khz*

##### **For FFT-Size:**

```
[FULLMAX(dl-config)]$ show fft_size  
fft_size 128fft
```

#### **4.5 Monitoring the BS / MS with FullMax NMS**

For monitoring the BS availability and performance it should be added to the NMS.

Login to NMS as an 'Admin'.

If the BS100 is located in a new tower:

On NMS menu select: Admin → New BS Tower

Type in tower name

Click 'Save'

On NMS menu select: Admin → New BS Sector

Type in a 'sector name'

Select the tower on which the sector is installed

Add the BS100 IP address

Click 'Save'

The BS100 availability and performance will now be monitored by the NMS

## 5 FullMax system operation

### 5.1 Operation Restrictions

- a) Changes or modifications not expressly approved by Full Spectrum Inc. could void the user's authority to operate the equipment.
- b) While operated as a Mobile (non fixed application), the MS400 equipment shall only be installed and operated with the antenna type, and antenna gain not more than that approved for this certification (as per the table below). The antenna will be installed with minimum separation distance between the antenna and the general public and between the antenna and the trained personnel during normal operation as stated in the table below for the respective antenna gains.
- c) An example of an antenna authorized for use with the MS400 while used in a mobile application is Laird # B8965C Mobile Load Coil Antenna with unity gain in the horizontal direction and with 5 dBi vertical gain.

Environment	Numeric Gain	Average Power (W)	Separation Distance (cm)	Power Density (mW/cm <sup>2</sup> )
General population/Uncontrolled	3.16	3.1	36	0.6
Occupational/Controlled	3.16	3.1	20	3
General population/Uncontrolled	20	3.1	91	0.6
Occupational/Controlled	20	3.1	41	3

- d) This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely

to cause harmful interference in which case the user will be required to correct the interference at his own expense.

## 5.2 BS100 Operation

FullMax BS CLI supports the following commands. The access to this commands are based on the user privileges. The Admin has the privilege to control and monitor all the information supported by the FullMax BS CLI commands, where as the operator has limited privileges. After the successful authentication, the user Is permitted access to the command line interface. Based on the user privileges, the help menu will be displayed to the user.

### 5.2.1 Main Group

When user logs in to the CLI it will lock to the default group which is the main group. When ever the user enters the Main group, CLI will display the main prompt .Help command in main group will show the commands supported in main group along with the group lock commands. Only commands related to main group can be executed in main group.

FullMax#

#### **ip-address**

System IP address.

Command	Purpose	Privileges
FullMax# set ip-address <ip-address>	Updates the system IP address.	A/O
FullMax# show ip-address	Displays the system IP address.	A/O

Example:

```
FullMax#set ip-address 10.60.4.42
```

Updated IP address: 10.60.4.42

```
FullMax# show ip-address
```

IP address: 10.60.4.56

#### **dhcp-config-server-ip**

DHCP server IP address.

Command	Purpose	Privileges
FullMax#set dhcp-config-server-ip <ip-address>	Allows the user to update the BS with the DHCP server IP address.	A/O
FullMax# show dhcp-config-server-ip	Allows the user to display the remote DHCP server IP address	A/O

Example:

```
FullMax# set dhcp-config-server-ip 10.60.1.11
```

```
Updated DHCP server IP 10.60.1.11
```

```
FullMax# show dhcp-config-server-ip
```

```
DHCP server IP 0.0.0.0
```

### **dhcp-config-remote-id**

DHCP remote ID.

<b>Command</b>	<b>Purpose</b>	<b>Privileges</b>
FullMax# set dhcp-config-remote-id <mac-address>	Allows the user to update the DHCP remote ID	A/O
FullMax# show dhcp-config-remote-id	Allows the user to displays the DHCP remote ID	A/O

Example:

```
FullMax# set dhcp-config-remote-id 0019D1:50CC62
```

```
Updated dhcp-config-remote-id 0019D1:50CC62
```

```
FullMax# show dhcp-config-remote-id
```

```
dhcp-config-remote-id 0019D1:50CC62
```

### **dhcp-config-circuit-id**

DHCP circuit ID

<b>Command</b>	<b>Purpose</b>	<b>Privileges</b>
FullMax# set dhcp-config-circuit-id	Allows the user to updates the DHCP circuit ID	A/O
FullMax# show dhcp-config-circuit-id	Allows the user to display the DHCP circuit ID	A/O

Example:

```
FullMax# set dhcp-config-circuit-id 3
```

```
Updated dhcp-config-circuit-id 3
```

```
FullMax# show dhcp-config-circuit-id
```

```
dhcp-config-circuit-id 0
```

## 5.2.2 mac-config

mac-config group defines the commands that monitors or updates the MAC configuration information. The user upon entering the mac-config command , locks into this particular group to execute commands only related to BS configuration.

```
FullMax#
FullMax# mac-config
FullMax(mac-config)#
```

Command	Purpose	Privileges
FullMax# mac-config	The command allows the user to lock into the group mac-config and execute all the commands related to the group.	A/O

The commands supported by the mac-config group are

### dcd-interval

The command describes the time between transmissions of DCD messages in milliseconds.

Command	Purpose	Privileges
FullMax(mac-config)# set dcd-interval <val>	The command allows the user to update the transmission time between to DCD messages.	A
FullMax(mac-config)#show dcd-interval	The command allows the user to find the time interval for transmission of consecutive DCD messages.	A/O

Example:

```
FullMax(mac-config)# set dcd-interval 30
Updated dcd-interval 30 milliseconds
```

```
FullMax(mac-config)# show dcd-interval
dcd-interval 30 milliseconds
```

### ucd-interval

The command describes the time between transmissions of UCD messages in milliseconds.

Command	Purpose	Privileges
FullMax(mac-config)# set ucd-interval <val>	The command allows the user to update the transmission time between to UCD messages.	A

FullMax(mac-config)# show ucd-interval	The command allows the user to find the time interval for transmission of consecutive UCD messages.	A/O
--	---	-----

Example:

```
FullMax(mac-config)# set ucd-interval 30
Updated ucd-interval 30 milliseconds
```

```
FullMax(mac-config)# show ucd-interval
ucd-interval 30 milliseconds
```

### dcd-transition

The number of frames from the end of the frame carrying the DCD message the BS shall wait after transmitting a DCD message with an incremented Configuration Change count before issuing a DL-MAP message referring to Downlink Burst profiles defined in that DCD message.

Command	Purpose	Privileges
FullMax(mac-config)# set dcd-transition <val>	This command allows the user to change the configuration count, the BS shall wait to transmit DL-MAP after transmitting the DCD message.	A
FullMax(mac-config)# show dcd-transition	The command allows the user to find the configuration count, the BS shall wait (no. of frames) to transmit DL-MAP after transmitting the DCD message.	A/O

Example:

```
FullMax(mac-config)# set dcd-transition 30
Updated dcd-transition 30 (MAC Frames)
```

```
FullMax(mac-config)# show dcd-transition
dcd-transition 30 (MAC Frames)
```

### ucd-transition

The number of frames from the end of the frame carrying the UCD message the BS shall wait after transmitting a UCD message with an incremented Configuration Change count before issuing a UL-MAP message referring to Uplink Burst profiles defined in that UCD message.

Command	Purpose	Privileges
FullMax(mac-config)# set ucd-transition <val>	This command allows the user to change the configuration count, the BS shall wait to transmit UL-MAP after transmitting the UCD message.	A
FullMax(mac-config)# show	The command allows the user to find the	A/O

ucd-transition	configuration count, the BS shall wait (no. of frames) to transmit UL-MAP after transmitting the UCD message.	
----------------	---	--

Example:

```
FullMax(mac-config)# set ucd-transition 30
Updated ucd-transition 30 (MAC Frames)
```

```
FullMax(mac-config)# show ucd-transition
ucd-transition 30 (MAC Frames)
```

### Dsx-rsp-retries

Number of Timeout Retries on DSA/DSC/DSD Responses.

Command	Purpose	Privileges
FullMax(mac-config)# set dsx-rsp-retries <val>	The command updates the value of the DSA/DSC/DSD responses retries timeout.	A
FullMax(mac-config)# show dsx-rsp-retries	The command displays the DSA/DSC/DSD responses retries timeout.	A/O

Example:

```
FullMax(mac-config)#set dsx-rsp-retries 6
Updated dsx-rsp-retries 6.
```

```
FullMax(mac-config)#show dsx-rsp-retries
dsx-rsp-retries 3
```

### Dsx-req-retries

Number of Timeout Retries on DSA/DSC/DSD Requests.

Command	Purpose	Privileges
FullMax(mac-config)# set dsx-req-retries <val>	The command updates the value of the DSA/DSC/DSD requests retries timeout.	A
FullMax(mac-config)#show dsx-req-retries	The command displays the DSA/DSC/DSD requests retries timeout.	A/O

Example:

```
FullMax(mac-config)#set dsx-req-retries 6000
Updated dsx-req-retries 6000 milliseconds.
```

```
FullMax(mac-config)#show dsx-req-retries
dsx-req-retries 3.
```



### T7-timeout

Wait for DSA/DSC/DSD Response Timeout in ms.

Command	Purpose	Privileges
FullMax (mac-config) # set t7-timeout <val>	The command updates the value of T7 timer, with in which the BS/SS should wait for a DSx response.	A
FullMax (mac-config) # show t7-timeout	The command displays the T7 timer, configured to which the BS/SS should wait for a DSx response.	A/O

Example:

```
FullMax(mac-config)#set t7-timeout 300
Updated t7-timeout 300 milliseconds.
```

```
FullMax(mac-config)# show t7-timeout
t7-timeout 300 milliseconds.
```

### T8-timeout

Wait for DSA/DSC/DSD Acknowledge Timeout in ms.

Command	Purpose	Privileges
FullMax(mac-config)# set t8-timeout <val>	The command updates the value of T8 timer, with in which the BS/SS should wait for a DSx acknowledge.	A
FullMax(mac-config)# show t8-timeout	The command displays the T8 timer, configured to which the BS/SS should wait for a DSx acknowledge.	A/O

Example:

```
FullMax(mac-config)# set t8-timeout 100
Updated t8-timeout 100 milliseconds.
```

```
FullMax(mac-config)# show t8-timeout
t8-timeout 100 milliseconds.
```

### T9-timeout

The time allowed between the BS sending a RNG-RSP to an SS, and receiving an SBC-REQ from that same SS.

Command	Purpose	Privileges
FullMax(mac-config)# set t9-timeout <val>	The command updates the T9 timer value that waits for an SBC-REQ from an MS to	A

	which it has sent a RNG-RSP.	
FullMax(mac-config)#show t9-timeout	The command displays the T9 Timer value configured in BS to wait for SBC-REQ from an MS to which the BS has sent the RNG-RSP.	A/O

Example:

```
FullMax(mac-config)#set t9-timeout 3200
Updated t9-timeout 3200 milliseconds
```

```
FullMax(mac-config)#show t9-timeout
t9-timeout 3200 milliseconds
```

### T10-timeout

The maximum time allowed to wait for a dynamic service transaction to end.

Command	Purpose	Privileges
FullMax(mac-config)#set t10-timeout <val>	The command updates the T10 timer value that waits for a dynamic service transaction end.	A
FullMax(mac-config)#show t10-timeout	The command displays the T10 timer value that waits for a dynamic service transaction end.	A/O

Example:

```
FullMax(mac-config)# set t10-timeout 1000
Updated t10-timeout 1000milliseconds
```

```
FullMax(mac-config)# show t10-timeout
T10-timeout 1000milliseconds
```

### T13-timeout

Time allowed for an SS, following the receipt of a REG-RSP message to send a TFTP-CPLT message to the BS.

Command	Purpose	Privileges
FullMax(mac-config)#set t13-timeout <val>	The command updates the T13 timer for which the BS waits to receive TFTP-CPLT message after sending the REG-RSP.	A
FullMax(mac-config)#show t13-timeout	The command displays the T13 timer configured at BS to wait for TFTP-CPLT message after sending the REG-RSP	A/O

Example:

```
FullMax(mac-config)# set t13-timeout 13000
```

```
Updated t13-timeout 13000 milliseconds
```

```
FullMax(mac-config)# show t13-timeout
t13-timeout 13000 milliseconds
```

### T17-timeout

Time allowed for SS to complete authorization and key exchange.

Command	Purpose	Privileges
FullMax(mac-config)#set t17-timeout <val>	The command updates the value of T17 timer, with in which the SS should complete the authorization and key exchange.	A
FullMax(mac-config)#show t17-timeout	The command displays the T17 timer, configured to which the SS should complete the authorization and key exchange.	A/O

Example:

```
FullMax(mac-config)#set t17-timeout 6000
Updated t17-timeout 6000 milliseconds.
```

```
FullMax(mac-config)#show t17-timeout
t17-timeout 6000 milliseconds.
```

### T22-timeout

Time to wait for an ARQ-reset.

Command	Purpose	Privileges
FullMax(mac-config)# set t22-timeout <val>	The command updates the T22 timer for which the BS waits for ARQ reset.	A
FullMax(mac-config)# show t22-timeout	The command displays the T22 timer configured at BS to wait for ARQ reset.	A/O

Example:

```
FullMax(mac-config)# set t22-timeout 200
Updated t22-timeout 200 milliseconds
```

```
FullMax(mac-config)# show t22-timeout
t22-timeout 200 milliseconds
```

### auto-sfid-enabled

The command defines whether BS is allowed to autonomously allocate SFIDs. If the BS is enabled with this feature, the BS can autonomously allocate SFIDs in configured SFID range. An SF is autonomously when it is not provisioned and may be initiated either by either BS or SS.

Command	Purpose	Privileges
FullMax(mac-config)# set auto-sfid-enabled {enable   disable }	This command allows the user to configure BS by allowing the BS to autonomously allocate SFIDs.	A
FullMax(mac-config)# show auto-sfid-enabled {enable   disable }	This command displays the user, whether the BS is enabled/disabled to allocate SFIDs.	A/ O

Example:

```
FullMax(mac-config)# set auto-sfid-enabled ENABLE
Updated auto-sfid-enabled ENABLED
```

```
FullMax(mac-config)# show auto-sfid-enabled
auto-sfid-enabled ENABLED
```

### min-auto-sfid-range

The command defines the minimum value of the SFID for the range allocated to support the creation of service flows autonomously.

Command	Purpose	Privileges
FullMax(mac-config)# set min-auto-sfid-range <val>	This command allows the user to define the minimum value of SFID to define the range in which the BS can autonomously create Service flows.	A
FullMax(mac-config)# show min-auto-sfid-range	The command displays the user, the minimum value of sfid from which the BS can create service flows autonomously.	A/O

Example:

```
FullMax(mac-config)# set min-auto-sfid-range 6200
Updated min-auto-sfid-range 6200
```

```
FullMax(mac-config)# show min-auto-sfid-range
Updated min-auto-sfid-range 6200
```

### max-auto-sfid-range

The command defines the maximum value of the SFID for the range allocated to support the creation of service flows autonomously.

Command	Purpose	Privileges
FullMax(mac-config)# set max-auto-sfid-range <val>	This command allows the user to define the maximum value of SFID range in which the BS can autonomously create Service flows.	A
FullMax(mac-config)# show	The command displays the user, the	A/O

max-auto-sfid-range	maximum value of sfid upto which the BS can create service flows autonomously.	
---------------------	--	--

Example:

```
FullMax(mac-config)# set max-auto-sfid-range 6300
Updated max-auto-sfid-range 6300
```

```
FullMax(mac-config)# show max-auto-sfid-range
max-auto-sfid-range 6300
```

### all

This command displays all the parametrs values for the mac-config group in CLI.

Command	Purpose	Privileges
FullMax(mac-config)#show all	The command displays the user, the value of all the mib parameters of tables in the group.	A/O

Example:

```
FullMax(mac-config)# show all

dcd-interval 5000 milliseconds
ucd-interval 5000 milliseconds
ucd-transition 0 (MAC Frames)
dcd-transition 0 (MAC Frames)
t9-timeout 8000 milliseconds
t13-timeout 0 milliseconds
t15-timer 0 milliseconds
t17-timeout 0 milliseconds
auto-sfid-enabled DISABLE
min-auto-sfid-range 0
max-auto-sfid-range 0
dsx-req-retries 3
dsx-rsp-retries 3
T7-timer 1000 milliseconds
t8-timer 300 milliseconds
t10-timer 3000 milliseconds
t22-timer 200 milliseconds
```

### 5.2.3 ul-config

The group defines the commands that monitors or updates the UCD channel attributes that define the characteristics of uplink channels. The user upon entering the ul-config command locks into the CLI to execute commands only related to BS uplink channel characteristics.

```
FullMax#
FullMax# ul-config
FullMax(ul-config)#
```

Command	Purpose
FullMax# ul-config	The command allows the user to lock into the group ul-config and execute all the commands related to the group.

### ct-based-reserve-timeout

The number of UL-MAPs to receive before contention-based reservation is attempted again for the same connection.

Command	Purpose	Privileges
FullMax(ul-config)# set ct-based-reserve-timeout <val>	The command allows the user to update number of UL-MAPs to receive before contention-based reservation is attempted again for the same connection.	A
FullMax(ul-config)#show ct-based-reserve-timeout	The command allows the user to display number of UL-MAPs to receive before contention-based reservation is attempted again for the same connection.	A/O

Example:

```
FullMax(ul-config)# set ct-based-reserve-timeout 30
Updated ct-based-reserve-timeout 30
FullMax(ul-config)# show ct-based-reserve-timeout
ct-based-reserve-timeout 30
```

### uplink-center-frequency

Uplink center frequency (in KHz).

Command	Purpose	Privileges
FullMax(ul-config)#set uplink-center-frequency <val>	The command allows the user to update the uplink center frequency.	A
FullMax(ul-config)#show uplink-center-frequency	The command allows the user to display the uplink center frequency.	A/O

Example:

```
FullMax (ul-config) # set uplink-center-frequency 30
updated uplink-center-frequency 30 KHz

FullMax (ul-config) # show uplink-center-frequency
uplink-center-frequency 30 KHz
```

## init-rng-codes

Number of initial ranging CDMA codes.

Command	Purpose	Privileges
FullMax(ul-config)# set init-rng-codes <val>	The command allows the user to update the Number of initial ranging CDMA codes.	A
FullMax(ul-config)#show init-rng-codes	The command allows the user to display the number of initial ranging CDMA codes.	A/O

Example:

```
FullMax(ul-config)# set init-rng-codes 10
Updated init-rng-codes 10
```

```
FullMax(ul-config)# show init-rng-codes
init-rng-codes 10
```

## periodic-rng-codes

Number of periodic ranging CDMA codes.

Command	Purpose	Privileges
FullMax(ul-config)# set periodic-rng-codes <val>	The command allows the user to update the Number of periodic ranging CDMA codes.	A
FullMax(ul-config)#show periodic-rng-codes	The command allows the user to display the number of periodic ranging CDMA codes.	A/O

Example:

```
FullMax(ul-config)# set periodic-rng-codes 20
Updated periodic-rng-codes 20
```

```
FullMax(ul-config)# show periodic-rng-codes
periodic-rng-codes 20
```

## bandwidth-request-codes

Number of bandwidth request codes.

Command	Purpose	Privileges
FullMax(ul-config)#set bandwidth-request- codes <val>	The command allows the user to update the number of bandwidth request codes.	A
FullMax(ul-config)#show	The command allows the user to display	A/O

bandwidth-request-codes	the number of bandwidth request codes.	
-------------------------	--	--

Example:

```
FullMax(ul-config)#set bandwidth-request-codes 20
Updated bandwidth-request-codes 20
```

```
FullMax(ul-config)#show bandwidth-request-codes
bandwidth-request-codes 20
```

### per-rng-backoff-start

Initial backoff window size for periodic ranging contention expressed as a power of 2

Command	Purpose	Privileges
FullMax(ul-config)#set per-rng-backoff-start <val>	The command allows the user to update the initial backoff window size for periodic ranging contention.	A
FullMax(ul-config)#show per-rng-backoff-start	The command allows the user to display the initial backoff window size for periodic ranging contention.	A/O

Example:

```
FullMax (ul-config) # set per-rng-backoff-start 4
Updated per-rng-backoff-start 4
```

```
FullMax (ul-config) # show per-rng-backoff-start
per-rng-backoff-start 4
```

### per-rng-backoff-end

Final backoff window size for periodic ranging contention, expressed as a power of 2

Command	Purpose	Privileges
FullMax(ul-config)# set per-rng-backoff-end <val>	The command allows the user to update the final backoff window size for periodic ranging contention.	A
FullMax(ul-config)# show per-rng-backoff-end	The command allows the user to display the final backoff window size for periodic ranging contention.	A/O

Example:

```
FullMax(ul-config)# set per-rng-backoff-end 8
Updated per-rng-backoff-end 8
```



```
FullMax(ul-config)# show per-rng-backoff-end
per-rng-backoff-end 8
```

### start-off-rng-codes

Indicates the starting number, of the group of codes used for uplink

Command	Purpose	Privileges
FullMax(ul-config)#set start-off-rng-codes <val>	The command allows the user to update the starting number of the codes used for uplink	A
FullMax(ul-config)#show start-off-rng-codes	The command allows the user to display the starting number of the codes used for uplink.	A/O

Example:

```
FullMax(ul-config)# set start-off-rng-codes 64
Updated start-off-rng-codes 64
```

```
FullMax(ul-config)# show start-off-rng-codes
start-off-rng-codes 64
```

### permutation-base

Determines the UL\_PermBase parameter for the sub carrier permutation to be used on this uplink channel.

Command	Purpose	Privileges
FullMax(ul-config)# set permutation-base <val>	The command allows the user to updates the UL_PermBase parameter for the sub carrier permutation to be used on this uplink channel.	A
FullMax(ul-config)# show permutation-base	The command allows the user to display the UL_PermBase parameter for the sub carrier permutation to be used on this uplink channel.	A/O

Example:

```
FullMax(ul-config)# set permutation-base 127
Updated permutation-base 127
```

```
FullMax(ul-config)#show permutation-base
permutation-base 8
```

### ul-alloc-sub-channel-bitmap

This is a bitmap describing the physical sub-channels allocated to the segment in the UL, when using the uplink PUSC permutation. The LSB of the first byte shall correspond to subchannel 0.

Command	Purpose	Privileges
FullMax(ul-config)# set ul-alloc-sub-channel-bitmap <val>	The command allows the used to update the ofdma subchannel allocation.	A
FullMax(ul-config)#show ul-alloc-sub-channel-bitmap	The command allows the user to display the ofdma subchannel allocation.	A/O

Example:

```
FullMax(ul-config)#set ul-alloc-sub-channel-bitmap
00110001
Updated ul-alloc-sub-channel-bitmap ff
```

```
FullMax(ul-config)# show ul-alloc-sub-channel-bitmap
ul-alloc-sub-channel-bitmap 00110001
```

### band-amc-alloc-threshold

Threshold of the maximum of the standard deviations of the individual bands CINR measurements over time to trigger mode transition from normal sub channel to Band AMC.

Command	Purpose	Privileges
FullMax(ul-config)# set band-amc-alloc-threshold <val>	The command allows the user to update the parameter for AMC band allocation Threshold.	A
FullMax(ul-config)#show band-amc-alloc-threshold	The command allows the user to display the parameter for AMC band allocation threshold.	A/O

Example:

```
FullMax(ul-config)# set band-amc-alloc-threshold 64
Updated band-amc-alloc-threshold 64 dB
```

```
FullMax(ul-config)# show band-amc-alloc-threshold
band-amc-alloc-threshold 64 dB
```

### band-amc-release-threshold

Threshold of the maximum of the standard deviations of the individual bands CINR measurements over time to trigger mode transition from Band AMC to normal subchannel.

Command	Purpose	Privileges
FullMax(ul-config)# set band-amc-release-threshold <val>	The command allows the user to update the parameter that defines the maximum threshold for AMC band.	A
FullMax(ul-config)#show band-amc-release-threshold	The command allows the user to display the parameter that defines the maximum threshold for AMC band.	A/O

Example:

```
FullMax(ul-config)# set band-amc-release-threshold 56
Updated band-amc-release-threshold 56 dB
```

```
FullMax(ul-config)# show band-amc-release-threshold
band-amc-release-threshold 56 dB
```

### **band-amc-alloc-timer**

Minimum required number of frames to measure the average and standard deviation for the event of Band AMC triggering.

Command	Purpose	Privileges
FullMax(ul-config)# set band-amc-alloc-timer <val>	The command allows the user to update the parameter that defines the minimum required number of frames to measure the average and standard deviation for the event of Band AMC triggering.	A
FullMax(ul-config)#show band-amc-alloc-timer	The command allows the user to display parameter that defines the minimum required number of frames to measure the average and standard deviation for the event of Band AMC triggering.	A/O

Example:

```
FullMax(ul-config)# set band-amc-alloc-timer 22
Updated band-amc-alloc-timer 22 Frames
```

```
FullMax(ul-config)# show band-amc-alloc-timer
band-amc-alloc-timer 22 Frames
```

### **band-amc-release-timer**

Minimum required number of frames to measure the average and standard deviation for the event of Band AMC triggering.

Command	Purpose	Privileges
FullMax(ul-config)# set band-amc-release-timer <val>	The command allows the user to update the parameter that defines the minimum required number of frames to measure the average and standard deviation for the event of Band AMC triggering.	A
FullMax(ul-config)#show band-amc-release-timer	The command allows the user to display parameter that defines the minimum required number of frames to measure the average and standard deviation for the event of Band AMC triggering.	A/O

Example:

```
FullMax(ul-config)#set band-amc-release-timer 22
Updated band-amc-release-timer 22 Frames
```

```
FullMax(ul-config)#show band-amc-release-timer
band-amc-release-timer 22 Frames
```

### band-amc-retry-timer

Backoff timer between consecutive mode transitions from normal subchannel to Band AMC when the previous request is failed.

Command	Purpose	Privileges
FullMax(ul-config)#set band-amc-retry-timer <val>	The command allows the user to update the parameter band AMC retry timer	A
FullMax(ul-config)#show band-amc-retry-timer	The command allows the user to display parameter band AMC retry timer.	A/O

Example:

```
FullMax(ul-config)# set band-amc-retry-timer 5
Updated band-amc-retry-timer 5
FullMax(ul-config)# show band-amc-retry-timer
band-amc-retry-timer 5
```

### band-stat-rep-max-period

Maximum period between refreshing the Band CINR measurement by the unsolicited REP-RSP.

Command	Purpose	Privileges
FullMax(ul-config)# set band-stat-rep-max-period <val>	The command allows the user to update the maximum period between refreshing the Band CINR measurement by the unsolicited REP-RSP.	A

FullMax(ul-config)# show band-stat-rep-max-period	The command allows the user to display the maximum period between refreshing the Band CINR measurement by the unsolicited REP-RSP.	A/O
---	--	-----

Example:

```
FullMax(ul-config)# set band-stat-rep-max-period 12
Updated band-stat-rep-max-period 12
```

```
FullMax(ul-config)# show band-stat-rep-max-period
band-stat-rep-max-period 12
```

### up-power-adj-step

MS specific up power offset adjustment step.

Command	Purpose	Privileges
FullMax(ul-config)#set band-stat-rep-max-period <val>	The command allows the user to update the maximum period between refreshing the Band CINR measurement by the unsolicited REP-RSP.	A
FullMax(ul-config)#show band-stat-rep-max-period	The command allows the user to display the maximum period between refreshing the Band CINR measurement by the unsolicited REP-RSP.	A/O

Example:

```
FullMax(ul-config)# set up-power-adj-step 12
Updated up-power-adj-step 12 (in 0.01 dB).
```

```
FullMax(ul-config)# show up-power-adj-step
up-power-adj-step 12 (in 0.01 dB).
```

### down-power-offset-adj-step

MS specific down power offset adjustment step.

Command	Purpose	Privileges
FullMax(ul-config)# set down-power-offset-adj-step <val>	The command allows the user to update the MS specific down power offset adjustment step.	A
FullMax ul-config)# show down-power-offset-adj-step	The command allows the user to display the MS specific down power offset adjustment step.	A/O

Example:

```
FullMax(ul-config)# set down-power-offset-adj-step 6
```

Updated down-power-offset-adj-step 6 (in 0.01 dB).

```
FullMax(ul-config)# show up-power-offset-adj-step
up-power-offset-adj-step 6 (in 0.01 dB).
```

### min-power-offset-adj

Minimum level of power offset adjustment.

Command	Purpose	Privileges
FullMax(ul-config)# set min-power-offset-adj <val>	The command allows the user to update the value of Minimum level of power offset adjustment.	A
FullMax(ul-config)#show min-power-offset-adj	The command allows the user to display the value of Minimum level of power offset adjustment.	A/O

Example:

```
FullMax(ul-config)# set min-power-offset-adj 2
Updated min-power-offset-adj 2 (in 0.01 dB).
FullMax(ul-config)# show min-power-offset-adj
min-power-offset-adj 2 (in 0.01 dB).
```

### max-power-offset-adj

Maximum level of power offset adjustment.

Command	Purpose	Privileges
FullMax(ul-config)#set max-power-offset-adj <val>	The command allows the user to update the value of maximum level of power offset adjustment.	A
FullMax(ul-config)#show max-power-offset-adj	The command allows the user to display the value of maximum level of power offset adjustment.	A/O

Example:

```
FullMax (ul-config) # set max-power-offset-adj 16
Updated max-power-offset-adj 16 (in 0.01 dB).

FullMax (ul-config) # show max-power-offset-adj
max-power-offset-adj 16 (in 0.01 dB).
```

### initial-rng-backoff-start

Initial backoff window size for initial ranging connection, expressed as a power of 2.

Command	Purpose	Privileges
FullMax(ul-config)# set intial-rng-backoff-start	The command allows the user to update the value of initial backoff window size for initial ranging connection.	A

<val>		
FullMax(ul-config)#show intial-rng-backoff-start	The command allows the user to display the value of initial backoff window size for initial ranging connection.	A/O

Example:

```
FullMax(ul-config)# set intial-rng-backoff-start 2
Updated intial-rng-backoff-start 2
```

```
FullMax(ul-config)# show intial-rng-backoff-start
intial-rng-backoff-start 2
```

### initial-rng-backoff-end

Final backoff window size for initial ranging connection, expressed as a power of 2.

Command	Purpose	Privileges
FullMax(ul-config)# set intial-rng-backoff-end <val>	The command allows the user to update the value of final backoff window size for initial ranging connection.	A
FullMax(ul-config)# show intial-rng-backoff-end	The command allows the user to display the value of final backoff window size for initial ranging connection.	A/O

Example:

```
FullMax(ul-config)# set intial-rng-backoff-end 8
Updated intial-rng-backoff-end 8
```

```
FullMax(ul-config)# show intial-rng-backoff-end
intial-rng-backoff-end 8
```

### bw-req-backoff-start

Initial backoff window size for contention bandwidth requests, expressed as a power of 2.

Command	Purpose	Privileges
FullMax(ul-config)# set bw-req-backoff-start <val>	The command allows the user to update the value of initial backoff window size for contention bandwidth requests.	A
FullMax(ul-config)#show bw-req-backoff-start	The command allows the user to display the value of initial backoff window size for contention bandwidth requests.	A/O

Example:

```
FullMax(ul-config)#set bw-req-backoff-start 4
Updated bw-req-backoff-start 4
```

```
FullMax(ul-config)#show bw-req-backoff-start
bw-req-backoff-start 4
```

### **bw-req-backoff-end**

Final backoff window size for contention bandwidth requests, expressed as a power of 2.

<b>Command</b>	<b>Purpose</b>	<b>Privileges</b>
FullMax(ul-config)# set bw-req-backoff-end <val>	The command allows the user to update the value of final backoff window size for contention bandwidth requests.	A
FullMax(ul-config)# show bw-req-backoff-end	The command allows the user to display the value of final backoff window size for contention bandwidth requests.	A/O

Example:

```
FullMax(ul-config)# set bw-req-backoff-end 8
Updated bw-req-backoff-end 4
FullMax(ul-config)# show bw-req-backoff-end
bw-req-backoff-end 4
```

### **all**

This command displays all the parameter values of tables supported in this group.

<b>Command</b>	<b>Purpose</b>	<b>Privileges</b>
FullMax(ul-config)# show all	The command allows the user to display the value of all the MIB parameters of tables supported in this group.	A/O

Example:

```
FullMax(ul-config)# show all
ct-based-reserve-timeout 200
uplink-center-frequency 200
init-rng-codes 0
periodic-rng-codes 50
bandwidth-request-codes 0
per-rng-backoff-start 20
permutation-base 0
ul-alloc-sub-channel-bitmap 3
band-amc-alloc-threshold 60 db
band-amc-release-threshold 0 db
```



```
band-amc-alloc-timer 255 Frames
band-amc-release-timer 0 Frames
```

### 5.2.4 dl-config

The group defines the commands that monitors or updates the DCD channel attributes that define the characteristics of downlink channel. The user upon entering the dl-config command locks into the CLI to execute commands only related to BS downlink channel characteristics.

```
FullMax#
FullMax# dl-config
FullMax(dl-config)#
```

Command	Purpose	Privileges
FullMax# dl-config	The command allows the user to lock into the group dl-config and execute all the commands related to the group.	A/O

### dl-channel-table

This command Displays the values of all the commands affiliated to dl-channel-table in the dl-config group.

Command	Purpose	Privileges
FullMax(dl-config)#show dl-channel-table	The command allows the user to display the DL Channel table MIB parameter values.	A/O

Example:

```
FullMax(dl-config)# show dl-channel-table
  bs-eirp 244 dBm
  downlink-center-frequency 244 khz
  bsid 00f401:0000f4
  hysteresis-margin 45 dB
  time-to-trigger 2 milliseconds
```

### frame-duration-code

The duration of the frame

Command	Purpose	Privileges
FullMax(dl-config)#set frame-duration-code	The command allows the user to update the OFDMA frame duration code.	A
FullMax(dl-config)#show frame-duration-code	The command allows the user to display the OFDMA frame duration code.	A/O

Example:

```
FullMax(dl-config)# set frame-duration-code 12
updated frame-duration-code 12

FullMax(dl-config)# show frame-duration-code
frame-duration-code 12
```

**bs-eirp**

The EIRP is the equivalent isotropic radiated power of the base station, which computed for a simple single-antenna transmitter.

Command	Purpose	Privileges
FullMax(dl-config)#set bs-eirp <val>	The command allows the user to update the BS EIRP parameter	A
FullMax(dl-config)#show bs-eirp	The command allows the user to display the BS EIRP parameter	A/O

Example:

```
FullMax(dl-config)# set bs-eirp 3770
Updated bs-eirp 3770 dBm

FullMax(dl-config)# show bs-eirp
bs-eirp 3770 dBm
```

**downlink-center-frequency**

Downlink center frequency (Khz).

Command	Purpose	Privileges
FullMax(dl-config)# set downlink-center-frequency <val>	The command allows the user to update the downlink center frequency.	A
FullMax(dl-config)#show downlink-center-frequency	The command allows the user to display the downlink center frequency.	A/O

Example:

```
FullMax (dl-config)# set downlink-center-frequency
11000
Updated downlink-center-frequency 11000 kHz

FullMax(dl-config)# show downlink-center-frequency
downlink-center-frequency 11000 kHz.
```

**bsid**

Defines the encoding of BSID. The BSID is a 6 byte number and follows the encoding rules of MacAddress textual convention, e.e as if it were transmitted least-significant bit first. The value should be displayed with 2 parts clearly separated by a colon e.g : 001DFF:00003A. The most significant part is representing the operator ID.

Command	Purpose	Privileges
FullMax(dl-config)# set bsid <val>	The command allows the user to update the BS ID	A
FullMax(dl-config)#show bsid	The command allows the user to display the BS ID	A/O

Example:

```
FullMax(dl-config)# set bsid 0A30A1:10FE11
Updated bsid 0A30A1:10FE11
```

```
FullMax(dl-config)# show bsid
bsid 0A30A1:10FE11
```

**bandwidth**

Channel Bandwidth in Khz

Command	Purpose	Privileges
FullMax(dl-config)# set bandwidth <val>	The command allows the user to updates the channel bandwidth.	A
FullMax(dl-config)#show bandwidth	The command allows the user to display the channel bandwidth.	A/O

Example:

```
FullMax(dl-config)# set bandwidth 400
Updated bandwidth 400 KHz
```

```
FullMax(dl-config)# show bandwidth
bandwidth 400 KHz
```

**Sampling-clock**

Sampling clock for the channel

Command	Purpose	Privileges
FullMax(dl-config)#set sampling-clock <val>	The command allows the user to updates the sampling clock.	A
FullMax(dl-config)#show sampling-clock	The command allows the user to display the sampling clock.	A/O

Example:

```
FullMax(dl-config)# set sampling-clock 12800
Updated sampling-clock 12800 KHz
```

```
FullMax(dl-config)# show sampling-clock
sampling-clock 12800 KHz
```

**filter-id**

The ID of the PHY filter to use for this channel

Command	Purpose	Privileges
FullMax(dl-config)# set filter-id <val>	The command allows the user to updates the filter id.	A
FullMax(dl-config)#show filter-id	The command allows the user to display the filter id.	A/O

Example:

```
FullMax(dl-config)# set filter-id 1
Updated filter-id 1
```

```
FullMax(dl-config)# show filter-id
filter-id 1
```

**fft\_size**

FFT size for OFDMA PHY

Command	Purpose	Privileges
FullMax(dl-config)# set fft_size <val>	The command allows the user to updates the fft_size.	A
FullMax(dl-config)#show fft_size	The command allows the user to display the fft_size	A/O

Example:

```
FullMax (dl-config)# set fft_size 512
Updated fft_size 512
```

```
FullMax (dl-config) # show fft_size
fft_size 512
```

**dl\_frame-duration**

This field indicates the number of OFDMA symbols allocated for downlink transmission in the OFDMA frame

Command	Purpose	Privileges
FullMax(dl-config)# set dl_frame-duration <val>	The command allows the user to updates the downlink frame duration	A

FullMax(dl-config)# show dl_frame-duration	The command allows the user to display the downlink frame duration	A/O
--	--	-----

Example:

```
FullMax (dl-config)# set dl_frame-duration 20
Updated dl_frame-duration 20 ofdma symbols
```

```
FullMax (dl-config) # show dl_frame-duration
dl_frame-duration 20 ofdma symbols
```

### **ul\_frame-duration**

This field indicates the number of OFDMA symbols allocated for uplink transmission in the OFDMA frame.

<b>Command</b>	<b>Purpose</b>	<b>Privileges</b>
FullMax(dl-config)# set ul_frame-duration <val>	The command allows the user to updates the uplink frame duration.	A
FullMax(dl-config)#show ul_frame-duration	The command allows the user to display the uplink frame duration.	A/O

Example:

```
FullMax (dl-config)# set ul_frame-duration 10
Updated ul_frame-duration 10 ofdma symbols
```

```
FullMax (dl-config) # show ul_frame-duration
ul_frame-duration 10 ofdma symbols
```

### **dl\_zones\_num**

Number of downlink zones for the BS

<b>Command</b>	<b>Purpose</b>	<b>Privileges</b>
FullMax(dl-config)#set dl_zones_num <val>	The command allows the user to update the number of downlink zones.	A
FullMax(dl-config)#show dl_zones_num	The command allows the user to display the number of downlink zones.	A/O

Example:

```
FullMax (dl-config)# set dl_zones_num 2
Updated dl_zones_num 2 zones
```

```
FullMax (dl-config) # show dl_zones_num
dl_zones_num 2 zones
```

**ul\_zones\_num**

Number of uplink zones for the BS

Command	Purpose	Privileges
FullMax(dl-config)# set ul_zones_num <val>	The command allows the user to updates the number of uplink zones	A
FullMax(dl-config)#show ul_zones_num	The command allows the user to display the number of uplink zones.	A/O

Example:

```
FullMax(dl-config)# set ul_zones_num 2
Updated ul_zones_num 2 zones
```

```
FullMax (dl-config)# show ul_zones_num
ul_zones_num 2 zones
```

**5.2.5 zone-config**

The group defines the commands that monitors or updates the downlink and uplink zones attributes. The user upon entering the zone-config command locks into the CLI to execute commands only related to BS downlink and uplink zones characteristics.

```
FullMax#
FullMax# zone-config
FullMax(zone-config)#
```

Command	Purpose	Privileges
FullMax# zone-config	The command allows the user to lock into the group zone-config and execute all the commands related to the group.	A/O

**dl-zone-id**

The downlink zone identifier ,the index of the first zone will always be 0

Command	Purpose	Privileges
FullMax(zone-config)#show dl-zone-id	The command allows the user to display the available DL zone ids.	A/O

Example:

```
FullMax(zone-config)#show dl-zone-id
dl-zone-id 0
dl-zone-id 1
```

```
dl-zone-id 2
```

### dl-zone-table

This table contains zone attributes that characterize a downlink zone

Command	Purpose	Privileges
FullMax(zone-config)# show dl_zone-table	The command allows the user to display the list of downlink zones.	A/O

```
FullMax(zone-config)# show dl-zone-table 2
dl-zone-stc 0
dl-perm-type pusc
dl-perm-base 17
dl-start-symbol 11
dl-all-sc YES
dl-cinr-threshold 2 dB
```

### dl-perm-type

Permutation type of this zone

Command	Purpose	Privileges
FullMax(zone-config)# set dl-perm-type <id> <val>	The command allows the user to update the downlink permutation type of zone <id>	A
FullMax(zone-config)# show dl-perm-type <id>	The command allows the user to display the permutation type of zone <id>	A/O

Example:

```
FullMax(zone-config)# set dl-perm-type 0 1
Updated dl-perm-type of zone 0 to pusc (1)

FullMax(zone-config)# show dl-perm-type 0
dl-perm-type of zone 0 is pusc (2)
```

### dl-zone-stc

Downlink zone

Command	Purpose	Privileges
FullMax(zone-config)# set dl-perm-zone <zone-id> <val>	The command allows the user to update the downlink permutation zone of zone <zone-id>	A
FullMax(zone-config)#show	The command allows the user to	A/O

dl-perm-zone <zone-id>	display the permutation zone of zone <zone-id>	
------------------------	--	--

Example:

```
FullMax(zone-config)# set dl-zone-stc 0 2
Updated dl-zone-stc of zone 0 to nonstc (2)
```

```
FullMax(zone-config)# show dl-zone-stc 0
dl-zone-stc of zone 0 is nonstc (2)
```

### dl-perm-base

Permutation base for this zone

Command	Purpose	Privileges
FullMax(zone-config)# set dl-perm-base <id> <val>	The command allows the user to update the downlink permutation base of zone <id>	A
FullMax(zone-config)# show dl-perm-base <id>	The command allows the user to display the permutation base of zone <id>	A/O

Example:

```
FullMax(zone-config)# set dl-perm-base 0 17
Updated dl-perm-base of zone 0 to 17
```

```
FullMax(zone-config)# show dl-perm-base
dl-perm-base of zone 0 is 17
```

### dl-start-symbol

Index of the starting symbol for this zone.

Command	Purpose	Privileges
FullMax(zone-config)# set dl-start-symbol <zone-id> <val>	The command allows the user to update the start-symbol downlink of <zone-id>	A
FullMax(zone-config)# show dl-start-symbol <zone-id>	The command allows the user to display the start-symbol of <zone-id>	A/O

Example:

```
FullMax (zone-config)# set dl-start-symbol 0 5
Updated dl-perm-type of zone 0 to 5
```

```
FullMax(zone-config)# show dl-start-symbol
dl-start-symbol of zone 0 is 5
```

### dl-all-sc



Use all subchannels.

1 (true): use\_all\_SC=1

2 (false): use\_all\_SC=0

Downlink zone parameters –

Command	Purpose	Privileges
FullMax (zone-config)#set dl-all-sc <zone-id> <val>	The command allows the user to update the downlink zone-all-subchannels parameter	A
FullMax(zone-config)#show dl-all-sc <zone-id>	The command allows the user to display the all subchannels parameter of <zone-id>	A/O

Example:

```
FullMax(zone-config)# set dl-all-sc 0 YES
Updated dl-all-sc of zone 0 to YES
```

```
FullMax (zone-config) # show dl-all-sc 0
dl-all-sc of zone 0 is YES
```

### dl-cinr-threshold

Threshold of the maximum of the standard deviations of the individual bands CINR measurements over time to trigger mode transition from zone 0 to this zone.

Command	Purpose	Privileges
FullMax(zone-config)# set dl-cinr-threshold <zone-id> <val>	The command allows the user to update the CINR threshold for <zone-id>	A
FullMax(zone-config)#show dl-cinr-threshold <zone-id>	The command allows the user to display the CINR threshold for <zone-id>	A/O

Example:

```
FullMax(zone-config)# set dl- cinr-threshold 0 2
Updated dl-cinr-threshold of zone 0 to 2 dB
FullMax(zone-config)# show dl-cinr-threshold 0
dl-cinr-threshold of zone 0 is 2 dB
```

### ul-Zone-id

The uplink zone identifier , The index of the first uplink zone will always be 0

Command	Purpose	Privileges
FullMax(zone-config)# show ul-zone-id	The command allows the user to display the ul zone indexes available	A/O

Example:

```
FullMax(zone-config)#show ul-zone-id
    ul-zone-id 0
    ul-zone-id 1
    ul-zone-id 2
```

## ul-zone-table

Displays the Uplink zones table

Command	Purpose	Privileges
FullMax(zone-config)#show ul_zone-table	The command allows the user to display the list of uplink zones.	A/O

Example:

```
FullMax(zone-config)# show ul-zone-table 0
    ul-perm-type for zone-id (0) : nonstc(2)
    ul-perm-base for zone-id (0) : 17
    ul-start-symbol for zone-id (0) : 11
    ul-alloc-bitmap for zone-id (0) : 3
    ul-cinr-threshold for zone-id (0) : 2 dB
```

## ul-perm-type

A zone permutation type

Command	Purpose	Privileges
FullMax(zone-config)#set ul-perm-type <id> <val>	The command allows the user to update the uplink permutation type of <zone-id>	A
FullMax(zone-config)#show ul-perm-type <id>	The command allows the user to display the permutation type of zone <zone-id>	A/O

Example:

```
FullMax(zone-config)# set ul-perm-type 0 2
    Updated ul-perm-type for zone-id(0): to non-stc (2)
FullMax(zone-config)# show ul-perm-type 0
    ul-perm-type for zone-id (0) : non-stc (2)
```

## ul-perm-base

Determines the Permutation Base parameter for the zone, permutation to be used on this uplink zone

Command	Purpose	Privileges
---------	---------	------------

FullMax(zone-config)# set ul-perm-base <id> <val>	The command allows the user to update the uplink permutation base of <zone-id>	A
FullMax(zone-config)# show ul-perm-base <id>	The command allows the user to display the permutation base of <zone-id>	A/O

Example:

```
FullMax(zone-config)# set ul-perm-base 0 17
Updated ul-perm-base for zone-id (0) to 17
FullMax (zone-config) # show ul-perm-base
ul-perm-base for zone-id (0) : 17
```

### ul-start-symbol

Index of the starting symbol for this zonehat segment

Command	Purpose	Privileges
FullMax(zone-config)# set ul-start-symbol <zone-id> <val>	The command allows the user to update the start-symbol uplink of <zone-id>	A
FullMax(zone-config)# show ul-start-symbol <zone-id>	The command allows the user to display the start-symbol of <zone-id>	A/O

Example:

```
FullMax (zone-config)# set ul-start-symbol 0 5
Updated ul-start-symbol for zone-id (0) to 5
FullMax (zone-config) # show ul-start-symbol
ul-start-symbol for-id zone-id(0) :5
```

### ul-alloc-bitmap

This is a bitmap describing the physical sub-channels allocated to the segment in the UL, when using the uplink PUSC permutation. The LSB of the first byte shall correspond to subchannel 0. For any bit that is not set, the corresponding subchannel shall not be used by the SS on that segment

Command	Purpose	Privileges
FullMax(zone-config)# set ul-alloc-bitmap <zone-id> <val>	The command allows the user to update the uplink zone – allocation bitmap parameter	A
FullMax(zone-config)# show ul-alloc-bitmap <zone-id>	The command allows the user to display the allocation bitmap parameter of zone <id>	A/O

Example:

```
FullMax(zone-config)# set ul-alloc-bitmap 0 010101
Updated ul-alloc-bitmap for zone-id (0) : to 010101
```

```
FullMax(zone-config) # show ul-alloc-bitmap 0
ul-alloc-bitmap of zone-id (0) : 01010101
```

### ul-cinr-threshold

This object is used to ensure that the write operation to multiple columns is guaranteed to be treated as atomic operation by agent.

Command	Purpose	Privileges
FullMax(zone-config)# set ul-cinr-threshold <zone-id> <val>	The command allows the user to update the CINR threshold for zone <id>	A
FullMax(zone-config)# show ul-cinr-threshold <zone-id>	The command allows the user to display the CINR threshold for zone <id>	A/O

Example:

```
FullMax(zone-config)# set ul-cinr-threshold 0 2
Updated ul-cinr-threshold for zone-id (0) to 2 dB
```

```
FullMax(zone-config)# show ul-cinr-threshold 0
ul-cinr-threshold for zone-id (0): 2 dB
```

### 5.2.6 bs-cap-config

The group defines the commands to configure the basic capabilities of BS. The user upon entering the bs-capability-config command locks into the CLI to execute commands only related to basic capabilities of BS.

```
FullMax#
FullMax# bs-cap-config
FullMax(bs-cap-config)#
```

Command	Purpose	Privileges
FullMax# bs-cap-config	The command allows the user to lock into the group bs-cap-config and execute all the commands related to the group.	A/O

all

#### Description

Command	Purpose	Privileges
FullMax(bs-cap-	The command allows the user to	A/O

onfig)#show all	display values of all the attributes of bs-cap-config group
-----------------	---

Example:

```
FullMax(bs-cap-config)# show all
  capability-ttg-transition-gap 34 microseconds
  capability-rtg-transition-gap 34 microseconds
  capability-pn-window-size 1000
  capability-number-of-ul-harq-channel 0
  capability-number-of-dl-harq-channel 0
```

### Capability-ttg-transition-gap

This parameter indicates the configured transition speed SSTTG for TDD and H-FDD SSs. The usage is defined by ss-transition-gap

Command	Purpose	Privileges
FullMax(bs-cap-config)# set capability-ttg-transitio-gap <val>	The command allows the user to update the TTG transition gap parameter	A
FullMax(bs-cap-config)# show capability-ttg-transitio-gap	The command allows the user to display the TTG transition gap parameter	A/O

Example:

```
FullMax(bs-cap-config)# set capability-ttg-transitio-gap
<val>
  Updated capability-ttg-transitio-gap
```

```
FullMax(bs-cap-config)# show capability-ttg-transitio-gap
  capability-ttg-transitio-gap
```

### Capability-rtg-transition-gap

This parameter indicates the configured transition speed SSTTG for TDD and H-FDD SSs. The usage is defined by ss-transition-gap.

Command	Purpose	Privileges
FullMax(bs-cap-config)# set capability-rtg-transition-gap <val>	The command allows the user to update the RTG transition gap.	A

FullMax(bs-cap-config)#show capability-rtg-transition-gap	The command allows the user to display the RTG transition gap.	A/O
---	--	-----

Example:

```
FullMax(bs-cap-config)# set capability-rtg-transition-gap
30
    Updated capability-rtg-transition-gap
```

```
FullMax (bs-cap-config)# show capability-rtg-transition-
gap
    capability-rtg-transition-gap 30
```

### 5.2.7 bs-burst-profile

The group defines the commands to burst profile commands of BS. The user upon entering the bs-burst-profile command locks into the CLI to execute commands only related to BS burst profile of BS.

```
FullMax#
    FullMax# bs-burst-profile
FullMax(bs-burst-profile)#
```

Command	Purpose	Privileges
FullMax#bs-burst-profile	The command allows the user to lock into the group bs-burst-profile and execute all the commands related to the group.	A/O

### uiuc-index

The uplink interval usage code indicates the uplink burst profile in the ucd message and is used along the index to identify the ucd-fec-code. Max uiuc -index range for OFDMA PHY <0- 10 >

Command	Purpose	Privileges
FullMax(bs-burst-Profile)# show uiuc-index	The command allows the user to display the Uplink interval usage codes indexes available in the MAC	A/O

Example:

```
FullMax(bs-burst-profile)# show uiuc-index
    uiuc-index 0
    uiuc-index 1
        uiuc-index 2
    uiuc-index 3
```

**ucd-fec-code-type**

Uplink FEC code type and modulation type

<b>Command</b>	<b>Purpose</b>	<b>Privileges</b>
FullMax (bs-burst-profile)# show dcd-fec-code-type <uiuc-index>	The command allows the user to display the different FEC and coding options for uplink burst profile.	A/O

Example:

```
FullMax(bs-burst-profile)# show ucd-fec-code-type 1
ucd-fec-code-type for uiuc-index (1) :
  qpskCc1Over2(0), qpskCc3Over4(1),
  sixteenQamCc1Over2(2), sixteenQamCc3Over4(3),
  sixtyFourQamCc1Over2(4), sixtyFourQamCc2Over3(5),
  sixtyFourQamCc3Over4(6), qpskBtc1Over2(7),
  qpskBtc3Over4Or2Over3(8), sixteenQamBtc3Over5(9),
  sixteenQamBtc4Over5(10),
  sixtyFourQamBtc2Over3Or5Over8(11),
  sixtyFourQamBtc5Over6Or4Over5(12), qpskCtc1Over2(13),
  reserved14(14), qpskCtc3Over4(15),
  sixteenQamCtc1Over2(16), sixteenQamCtc3Over4(17),
  sixtyFourQamCtc1Over2(18), sixtyFourQamCtc2Over3(19),
  sixtyFourQamCtc3Over4(20), sixtyFourQamCtc5Over6(21),
  qpskZtCc1Over2(22), qpskZtCc3Over4(23),
  sixteenQamZtCc1Over2(24), sixteenQamZtCc3Over4(25),
  sixtyFourQamZtCc1Over2(26),
  sixtyFourQamZtCc2Over3(27),
  sixtyFourQamZtCc3Over4(28), qpskLdpc1over2(29),
  qpskLdpc2over3A(30), qpskLdpc3over4A(31),
  sixteenQamLdpc1over2(32), sixteenQamLdpc2over3A(33),
  sixteenQamLdpc3over4A(34), sixtyFourQamLdpc1over2(35),
  sixtyFourQamLdpc2over3A(36),
  sixtyFourQamLdpc3over4A(37), qpskLdpc2over3B(38),
  qpskLdpc3over4B(39), sixteenQamLdpc2over3B(40),
  sixteenQamLdpc3over4B(41),
  sixtyFourQamLdpc2over3B(42),
  sixtyFourQamLdpc3over4B(43), qpskCcOptIntv1over2(44),
  qpskCcOptIntv3over4(45),
  sixteenQamCcOptIntv1over2(46),
  sixteenQamCcOptIntv3over4(47),
  sixtyFourQamCcOptIntv2over3(48),
  sixtyFourQamCcOptIntv3over4(49), qpskLdpc5over6(50),
  sixteenQamLdpc5over6(51), sixtyFourQamLdpc5over6(52)
```

diuc-index

The Downlink interval usage code indicates the uplink burst profile in the ucd message and is used along the index to identify the Dcd-fec-code. Max diuc -index range for OFDMA PHY <0- 12 >

Command	Purpose	Privileges
FullMax(bs-burst-Profile)# show diuc-index	The command allows the user to display the Downlink interval usage codes indexes available in the MAC	A/O

Example:

```
FullMax(bs-burst-profile)# show diuc index
    diuc-index 0
    diuc-index 1
    diuc-index 2
    diuc-index 3
```

### dcd-fec-code-type

Displays FEC and coding options for downlink burst profiles

Command	Purpose	Privileges
FullMax (bs-burst-profile)# show dcd-fec-code-type <diuc-index>	The command allows the user to display the different FEC and coding options for downlink burst profile.	A/O

Example:

```
FullMax (bs-burst-profile) # show dcd-fec-code-type 2
dcd-fec-code-type for diuc-index (2) is
qpskCc1Over2(0), qpskCc3Over4(1),
sixteenQamCc1Over2(2), sixteenQamCc3Over4(3),
sixtyFourQamCc1Over2(4), sixtyFourQamCc2Over3(5),
sixtyFourQamCc3Over4(6), qpskBtc1Over2(7),
qpskBtc3Over4Or2Over3(8), sixteenQamBtc3Over5(9),
sixteenQamBtc4Over5(10),
sixtyFourQamBtc2Over3Or5Over8(11),
sixtyFourQamBtc5Over6Or4Over5(12), qpskCtc1Over2(13),
reserved14(14), qpskCtc3Over4(15),
sixteenQamCtc1Over2(16), sixteenQamCtc3Over4(17),
sixtyFourQamCtc1Over2(18), sixtyFourQamCtc2Over3(19),
sixtyFourQamCtc3Over4(20), sixtyFourQamCtc5Over6(21),
qpskZtCc1Over2(22), qpskZtCc3Over4(23),
sixteenQamZtCc1Over2(24), sixteenQamZtCc3Over4(25),
sixtyFourQamZtCc1Over2(26),
sixtyFourQamZtCc2Over3(27),
```



sixtyFourQamZtCc3Over4(28), qpskLdpc1over2(29),  
 qpskLdpc2over3A(30), qpskLdpc3over4A(31),  
 sixteenQamLdpc1over2(32), sixteenQamLdpc2over3A(33),  
 sixteenQamLdpc3over4A(34), sixtyFourQamLdpc1over2(35),  
 sixtyFourQamLdpc2over3A(36),  
 sixtyFourQamLdpc3over4A(37), qpskLdpc2over3B(38),  
 qpskLdpc3over4B(39), sixteenQamLdpc2over3B(40),  
 sixteenQamLdpc3over4B(41),  
 sixtyFourQamLdpc2over3B(42),  
 sixtyFourQamLdpc3over4B(43), qpskCcOptIntv1over2(44),  
 qpskCcOptIntv3over4(45),  
 sixteenQamCcOptIntv1over2(46),  
 sixteenQamCcOptIntv3over4(47),  
 sixtyFourQamCcOptIntv2over3(48),  
 sixtyFourQamCcOptIntv3over4(49), qpskLdpc5over6(50),  
 sixteenQamLdpc5over6(51), sixtyFourQamLdpc5over6(52)

### 5.2.8 bs-ss-action

The bs-ss-action group define the commands that monitors or updates the actions specified for SS. The user upon entering the bs-ss-action command locks into the CLI to execute commands only related to BS private mib.

```
FullMax#
FullMax#bs-ss-action
FullMax(bs-ss-action)#
```

Command	Purpose	Privileges
FullMax#bs-ss-action	The command allows the user to lock into the group bs-ss-action and execute all the commands related to the group.	A/O

### reset-ss

Reset action performed on SS.

Command	Purpose	Privileges
FullMax(bs-ss-action)#set reset-ss <mac-address>	The command allows the user to perform reset action on SS	A

Example:

```
FullMax(bs-ss-action)# set reset-ss 1 0A1234:5678AB
Reset-ss message sent to 0A1234:5678AB
```

### abort-ss

Abort action performed on SS.

Command	Purpose	Privileges
FullMax(bs-ss-action)#set abort-ss <abort-op> <mac-address> [-d:dl-freq] [-u:up-id]	The command allows the user to perform the abort action on SS	A/O

```
FullMax(bs-ss-action)# set abort-ss 1 <mac-address>
    abort-ss message sent to <mac-address>
FullMax(bs-ss-action)# set abort-ss 2 0A1234:5678AB
-d:210000
    abort-ss message with downlink frequency override
210000 KHz sent to 0A1234:5678AB
```

### de-reg-ss

De-registration action on SS

Command	Purpose	Privileges
FullMax(bs-ss-action)# set abort-ss <abort-op> <mac-address> [-d:dl-freq] [-u:up-id]	The command allows the user to perform the abort action on SS	A/O

```
FullMax(bs-ss-action)# set de-reg-ss 0A1234:5678AB 2    De-
reg-ss message sent to 0A1234:5678AB with limited
transmission code (2)
FullMax(bs-ss-action)# set de-reg-ss 0A1234:5678AB 1
    De-reg-ss message sent to 0A1234:5678AB with no
transmission code (1)
```

### 5.2.9 ss-registered

The ss-registered group defines the commands that monitors or updates the standard mib related information. The user upon entering the ss-registered command locks into the CLI to execute commands only related to SS registered table mib.

```
FullMax#
    FullMax# ss-registered
FullMax ss-registered)#
```

Command	Purpose	Privileges
FullMax# ss-registered	The command allows the user to lock into the group ss-registered and execute all the commands related to the group.	A/O

### ss-mac-addresss

The MAC address of SS is received in the RNG-REQ message when SS registers ,this MAC address is entered in to the Bs Registered Ss Table.

Command	Purpose	Privileges
FullMax(ss-registered)# show registered-ss-mac-address	The command allows the user display all the SS MAC addresses.	A/O

Example:

```
FullMax(ss-registered)# show registered-ss-mac-address
Registered SS MAC Addresses are:
registered-ss-mac-address 1 001BFE:01020A
registered-ss-mac-address 2 001BAA:01030F
registered-ss-mac-address 3 001BEF:01F01A
```

### max-tx-power-qpsk

SS's Maximum available power for QPSK in dBm. This parameter is only applicable to systems supporting the SCA, OFDM or OFDMA PHY.

Command	Purpose	Privileges
FullMax(ss-registered)# show max-tx-power-qpsk <Key>	The command allows the user display SS's Maximum available power for QPSK in dBm. Here key is SS's MAC address.	A/O

Example:

```
FullMax(ss-registered)# show max-tx-power-qpsk
001BFE:01020A
max-tx-power-qpsk 27 dBm
```

### max-tx-power-16qam

SS's Maximum available power for 16-QAM constellations in dBm. This parameter is only applicable to systems supporting the SCA, OFDM or OFDMA PHY.

Command	Purpose	Privileges
FullMax(ss-registered)# show max-tx-power-16qam <key>	The command allows the user display SS's Maximum available power for 16QAM in dBm. Here key is SS's MAC address.	A/O

Example:

```
FullMax(ss-registered)# show max-tx-power-16qam
001BAA:01030F
    max-tx-power-16qam 20 dBm
```

### max-tx-power-64qam

SS's Maximum available power for 64-QAM constellations in dBm. This parameter is only applicable to systems supporting the SCA, OFDM or OFDMA PHY.

Command	Purpose	Privileges
FullMax(ss-registered)# show max-tx-power-64qam <key>	The command allows the user display SS's Maximum available power for 64QAM in dBm. Here key is SS's MAC address.	A/O

Example:

```
FullMax(ss-registered)# show max-tx-power-64qam
001BEF:01F01A
    max-tx-power-64qam 25 dBm
```

### 5.2.10 ss-ip

The bs-ss-ip group define the commands that monitors or updates the BsSsIp related information. The user upon entering the ss-ip command locks into the CLI to execute commands only related to BsSsIpTable defined in the private mib.

```
FullMax#
    FullMax# ss-ip
FullMax(ss-ip)#
```

Command	Purpose	Privileges
FullMax# ss-ip	The command allows the user to lock into the group ss-ip and execute all the commands related to the group.	A/O

### ss-ip-table

This table contains the IP configuration information of Subscriber station's as set by DHCP Relay function in the Base Station.

Command	Purpose	Privileges
FullMax(ss-ip)#show ss-ip-table	The command allows the user display the MS IP table.	A/O

Example:

```
FullMax (bs-private) # show ss-ip-table 001BFE:01020A
    ss-mac-address 001BFE:01020A
```

```
ss-ip-address 192.168.0.1
ss-subnet-mask 192.168.0.0
```

### ss-mac-address

The MAC address of SS is received from the RNG-REQ message. When SS registers, this MAC address is entered into the table, and used as the identifier to the SS.

Command	Purpose	Privileges
FullMax (ss-ip) # show ss-mac-address	The command allows the user display the MS MAC address of all MS.	A/O

Example:

```
FullMax (ss-ip) # show ss-mac-address
ss-mac-address 00A1B3:01AB03
ss-mac-address 00A1B3:010203
ss-mac-address 00A1B3:010405
```

### ss-ip-address

The subnet mask address of SS as received from BS DHCP Relay. When SS Registers, this IP address is entered into the table, and used as the identifier to the SS.

Command	Purpose	Privileges
FullMax (ss-ip) # show ss-ip-address	The command allows the user display the IP address.	A/O

Example:

```
FullMax(ss-ip)# show ss-ip-address 00A1B3:01AB03
ss-ip-address 192.168.0.1
```

### ss-subnet-mask

The IP address of SS is received from the DHCP-Relay in the BS when SS register, this IP address is entered into the table, and used as the identifier to the SS.

Command	Purpose	Privileges
FullMax(ss-ip) #show ss-ip-address	The command allows the user display the subnet mask of all the MS.	A/O

Example:

```
FullMax(ss-ip) # show ss-subnet-mask 00A1B3:01AB03
ss-subnet-mask 192.168.0.0
```

### 5.2.11 device

The device group define the commands that monitors or updates the device related information in the private MIB. The user upon entering the device command locks into the CLI to execute commands only related to device.

```
FullMax#
```

```
    FullMax# device
```

```
FullMax(device)#
```

Command	Purpose	Privileges
FullMax# device	The command allows the user to lock into the group device and execute all the commands related to the group.	A/O

### type

Type of device (BS/MS/SS).

Command	Purpose	Privileges
FullMax(device) # show type	The command allows the user display the device type (BS/MS/SS)	A/O

Example:

```
FullMax(device)# show type
    type BS
```

### gpos

The geographical position of the device , i.e. the real number describing the longitude and latitudes encoded as a printable string. Longitude - the precision is within the range -90..90 degrees. Positive numbers indicate locations north of the equator.

Latitude - The precision is within the range -180..180 degrees. Positive numbers indicate locations east of the prime meridian

Command	Purpose	Privileges
FullMax(device)# set gpos <val>	The command allows the user to update the geographical position of the device	A
FullMax(device)# show gpos	The command allows the user display geographical position of the device.	A/O

Example:

```
FullMax(device)# set gpos 64.000 88.3400
    updated gpos 64.000 88.3400
```

```
FullMax(device) # show gpos
gpos 64.000 88.3400
```

## boot-time

The absolute time of last device boot up.

Command	Purpose	Privileges
FullMax(device) # show boot-time	The command allows the user display the device boot up time.	A/O

Example:

```
FullMax(device) # show boot-time
boot-time 230404:18052008
```

## commit-save

Setting this object to TRUE causes the device to write all configuration changes in FLASH memory. On next boot the changes will be relevant. If this operation does not occur, configuration changes will not be maintained through reset Reading this object always returns FALSE

Command	Purpose	Privileges
FullMax(device) # set commit-save <val>	The command allows the user update the commit-save to TRUE .	A
FullMax(device) # show commit-save	The command allows the user display the commit-save	A/O

Example:

```
FullMax(device) # set commit-save TRUE
Updated commit-save TRUE
FullMax(device) # show commit-save
commit-save FALSE
```

## gps-card

GPS card availability in the device.

Command	Purpose	Privileges
FullMax(device) # sho w gps-card	The command allows the user display the availability of GPS card in the device.	A/O

Example:

```
FullMax(device) # show gps-card
gps-card YES
```

## bpc-hw-version

Hardware version of baseband processor card.

Command	Purpose	Privileges
FullMax(device)#show bpc-hw-version	The command allows the user display the hardware version of baseband processor card.	A/O

Example:

```
FullMax(device)# show bpc-hw-version
bpc-hw-version 1.10
```

## afe-hw-ver

Hardware version of the Analog Front End (AFE)

Command	Purpose	Privileges
FullMax(device)#show afe-hw-ver	The command allows the user display the hardware version of AFE.	A/O

Example:

```
FullMax(device)# show afe-hw-ver
device-afe-hw-version 1.3
```

## afe-sw-ver

Software version of the Analog Front End (AFE)

Command	Purpose	Privileges
FullMax(device)#show afe-sw-ver	The command allows the user display the software version of AFE.	A/O

Example:

```
FullMax(device)# show afe-sw-ver
device-afe-sw-version 1.3
```

## build-sw-ver

Software version of the device general embedded software

Command	Purpose	Privileges
FullMax(device)#show build-sw-ver	The command allows the user display the software version of the software build	A/O

Example:



```
FullMax(device)# show build-sw-ver
    build-sw-version
```

### 5.2.12 Measurements

The Measurement group defines the commands that monitors or updates the measurement related information in the private MIB. The user upon entering the device command locks into the CLI to execute commands only related to Measurement.

```
FullMax#
FullMax# measurement
FullMax(measurement)#
```

#### Temperature

Temperature degree in Celsius

Command	Purpose	Privileges
FullMax(measurement)#show temperature	The command allows the user display temperature of the device.	A/O
FullMax(measurement)#track [-rN] [-tN] temperature	The command allows the user to display real time temperature sample of the device.	A/O

Example:

```
FullMax(measurement)# show temperature
    Temperature 45 degrees-Celsius
FullMax(measurement)# track -r3 -i2 temperature
    Temperature 45 degrees-Celsius
    Temperature 46 degrees-Celsius
    Temperature 45 degrees-Celsius
```

#### Voltage

Voltage measurement

Command	Purpose	Privileges
FullMax(measurement)# show voltage	The command allows the user display voltage of the device.	A/O
FullMax(measurement)# track [-rN] [-tN] voltage	The command allows the user to display multiple voltage samples of the device.	A/O

Example:

```
FullMax (measurement) # show voltage
    Voltage 3 volts
FullMax (measurement) # track -r3 -i10 voltage
```

Voltage 3 volts  
 Voltage 3 volts  
 Voltage 3 volts

## Current

### Current measurements

Command	Purpose	Privileges
FullMax(measurement) # show current	The command allows the user display the current in the device.	A/O
FullMax(measurement) # track [-rN] [-tN] current	The command allows the user to display multiple current measurements in the device.	A/O

#### Example:

```
FullMax (measurement) # show current
    Current 2 amp
FullMax (measurement) # track -r4 current
    Current 2 amp
    Current 2 amp
    Current 2 amp
```

## Tx-power

### Transmit power

Command	Purpose	Privileges
FullMax(measurement) )# show tx-power	The command allows the user display the transmit power.	A/O
FullMax(measurement) # track [-rN] [-tN] tx- power	The command allows the user to display multiple transmit power measurements.	A/O

#### Example:

```
FullMax (measurement) # show tx-power
    Tx-power 45 dBm
FullMax (measurement) # track -r2 tx-power
    Tx-power 45 dBm
    Tx-power 43 dBm
```

## Rx-power

### Receive power

Command	Purpose	Privileges
FullMax(measurement) #	The command allows the user display	A/O

show rx-power	the receive power.	
FullMax(measurement)# track [-rN] [-tN] rx- power	The command allows the user to display multiple receive power measurements.	A/O

Example:

```
FullMax(measurement)# show rx-power
rx-power 45 dBm
FullMax(measurement)# track -r2 -i2 rx-power
rx-power 45 dBm
rx-power 45 dBm
```

## afe-temperature

Read the temperature from the Analog Front End.

Command	Purpose	Privileges
FullMax(measurement)# show afe-temperature	The command allows the user display temperatue from Analog Front End.	A/O
FullMax(measurement)# track [-rN] [-tN] afe-temperature	The command allows the user to track the temperatue from Analog Front End.	A/O

Example:

```
FullMax(measurement)# show afe-temperatue
afe-temperature 27 celcius
FullMax(measurement)# track -r4 -i30 afe-temperatue
afe-temperature 27 celcius
afe-temperature 27 celcius
afe-temperature 27 celcius
afe-temperature 27 celcius
```

## afe-rssi

Read the Receive Signal Strength Indicator (RSSI) from the Analog Front End.

Command	Purpose	Privileges
FullMax(measurement)#show afe-rssi	The command allows the user display RSSI from Analog Front End.	A/O
FullMax(measurement)#track [-rN] [-tN] afe-rssi	The command allows the user to track RSSI from Analog Front End.	A/O

Example:

```
FullMax(measurement)# show afe-rssi
afe-rssi 80 dBm
FullMax(measurement)# track -r3 -i30 afe-rssi
```

```
afe-rssi 80 dBm
afe-rssi 80 dBm
afe-rssi 80 dBm
```

### 5.2.13 pkm-config

#### pkm-config

The Pkm-Config group defines the commands that monitors or updates the Private Key Management Version 2 Related information in the Standard MIB .The user upon entering the pkm-config command locks into this group to execute commands only related to this group.

```
FullMax#
    FullMax# pkm-config
FullMax(pkm-config)
```

Command	Purpose	Privileges
FullMax# pkm -config	The command allows the user to lock to pkm-config group and display the pkm -config prompt.	A/O

```
FullMax# pkm-config
```

#### pkm-v2-config-table

This table conatins the configuration of the pkm attributes that are needed to pkm operation.

Command	Purpose	Privileges
FullMax(pkm-config)# show pkm-v2-config-table	The command allows the user display all the Pkm V2 Config table parameter values.	A/O

Example:

```
FullMax(pkm-config)# show pkm-v2-config-table
pkm-pre-handshake-lifetime 10 second
pkm-lifetime 3600 seconds
sa-tek-challenge-timeout 10000 milliseconds
max-sa-tek-challenge 3
max-sa-tek-request 3
```

### pkm-pmk-pre-handshake-lifetime

Defines the PkmPmkPrehandshakeLifetime.

Command	Purpose	Privileges
FullMax (pkm-config)#set pkm-pmk-pre-handshake-lifetime <val>	The command allows the user update PkmPmkPrehandshakeLifetime.	A/O
FullMax (pkm-config)#show pkm-pmk-pre-handshake-lifetime	The command allows the user to display PkmPmkPrehandshakeLifetime value	A/O

Example:

```
FullMax (pkm-config)# set pkm-pmk-pre-handshake-lifetime
11
Updated pkm-pmk-pre-handshake-lifetime 11 second
```

```
FullMax (pkm-config)# show pkm-pmk-pre-handshake-lifetime
pkm-pre-handshake-lifetime 10 second
```

### pkm-pmk-lifetime

Defines the PMK life time

Command	Purpose	Privileges
FullMax (pkm-config)#set pkm-pmk-lifetime <val>	The command allows the user to update pkm-pmk-lifetime value	A
FullMax (pkm-config)#show pkm-pmk-lifetime	The command allows the user display pkm-pmk-lifetime.	A/O

Example:

```
FullMax (pkm-config) # set pkm-pmk-lifetime 3300
Updated pkm-pmk-lifetime 3300 seconds
FullMax (pkm-config) # show pkm-pmk-lifetime 3600 seconds
pkm-pmk-lifetime 3300 seconds
```

### sa-challenge-timeout

This object defines the timeout value for SA-TEK challenge Retransmission.

Command	Purpose	Privileges

FullMax(pkm-config)#set sa-challenge-timeout <val>	The command allows the user to update sa-challenge-timeout.	A
FullMax(pkm-config)#show sa-challenge-timeout	The command allows the user display sa-challenge-timeout.	A/O

Example:

```
FullMax(pkm-config) # set sa-challenge-timeout 666
    updated value of sa-tek-challenge-timeout 666
millisecond
```

```
FullMax(pkm-config) # show sa-challenge-timeout
    sa-tek-challenge-timeout 10000 milliseconds
```

### max-sa-tek-challenge

This object defines the maximum number of SA-TEK-Challenge Transmissions

Command	Purpose	Privileges
FullMax(pkm-config)#set max-sa-tek-challenge <val>	The command allows the user to update sa-challenge-timeout.	A
FullMax(pkm-config)#show max-sa-tek-challenge	The command allows the user display max-sa-tek-challenge.	A/O

Example:

```
FullMax(pkm-config)# set max-sa-tek-challenge 2
    updated value of max-sa-tek-challenge 2
```

```
FullMax(pkm-config)# show max-sa-tek-challenge
    max-sa-tek-challenge 3
```

### max-sa-tek-request

This object defines the maximum number of SA-TEK -Request retransmission.

Command	Purpose	Privileges
FullMax(pkm-config)#set max-sa-tek-request <val>	The command allows the user to update max-sa-tek-request.	A
FullMax(pkm-config)#show max-sa-tek-request	The command allows the user to display max-sa-tek-request.	A/O

Example:

```
FullMax(pkm-config)# set max-sa-tek-request 2
    updated value of max-sa-tek-request 2
FullMax(pkm-config)# show max-sa-tek-request
    max-sa-tek-request 3
```

## all

The command allows the user to display all the parameters of the tables in this group.

Command	Purpose	Privileges
FullMax (pkm-config) #show all	The command allows the user to display all the parameters values of the in this group.	A/O

```
FullMax(pkm-config) # show all
    pmk-pre-handshake-lifetime 10 second
    pmk-lifetime 3600 seconds
    sa-tek-challenge-timeout 10000 milliseconds
    max-sa-tek-challenge 3
    max-sa-tek-request 3
```

### 5.2.14 bs-private

```
FullMax#
    FullMax# bs-private
FullMax(bs-private)#
```

Command	Purpose	Privileges
FullMax#bs-private	The command allows the user to lock to bs-private group and display the bs-private prompt.	A/O

### bs-rx-amc-count-table

This table contains statistical information that can be used to characterize the adaptive modulation and coding performance in the uplink.

Command	Purpose	Privileges
FullMax(bs-private)# show bs-rx-amc-count-table <uiuc-index>	This command shows all the parameters and their values for AmcCountTable	A/O

Example:

```
FullMax(bs-private)# show bs-rx-amc-count-table 1
    bs-rx-octets for uiuc index 1 is 0
```

```
bs-rx-packets for uiuc index 1 is 0
bs-tx-erroredpackets for uiuc index 1 is 0
```

### bs-rx-uiuc-index

The Uplink Interval Usage Code indicates the uplink burst profile in the UCD message, and is used along with ifIndex to identify an entry in the wmanIf2BsOfdmaUcdBurstProfileTable.

Command	Purpose	Privileges
FullMax(bs-private)#show bs-rx-uiuc-index	Show all the Uplink Interval Usage Code index available.A/O	A/O

Example:

```
FullMax(bs-private)# show bs-rx-uiuc-index
uiuc indices are
bs-ul-uiuc-index: 1
bs-ul-uiuc-index: 2
bs-ul-uiuc-index: 3
bs-ul-uiuc-index: 4
bs-ul-uiuc-index: 5
bs-ul-uiuc-index: 6
bs-ul-uiuc-index: 7
bs-ul-uiuc-index: 8
bs-ul-uiuc-index: 9
bs-ul-uiuc-index: 10
```

### bs-rx-octets

This object counts the number of octets received in the uplink using the uplink burst profile indexed by wmanPriBsRxUiucIndex.

Command	Purpose	Privileges
FullMax(bs-private)#show bs-rx-octets <uiuc-index>	Show the no of octet received in the uplink for the given index	A/O

Example:

```
FullMax(bs-private)# show bs-rx-octets 1
bs-rx-octets for uiuc index 1 is 0
```

### bs-rx-packets

This object counts the number of packets received in the uplink using the uplink burst profile indexed by UiucIndex

Command	Purpose	Privileges
FullMax(bs-private)#show bs-	Shows the number of packets received in the uplink for the	A/O



rx-packets <uiuc-index>	given index	
-------------------------	-------------	--

Example:

```
FullMax(bs-private)# show bs-rx-packets 1
bs-rx-packets for uiuc index 1 is 0
```

### bs-rx-erroredPackets

This object counts the number of errored packets received in the uplink using the uplink burst profile indexed by UiucIndex.

Command	Purpose	Privileges
FullMax(bs-private)#show bs-rx-erroredPackets <uiuc-index>	Shows the number of errored packets received in the uplink for the given index	A/O

Example:

```
FullMax(bs-private)# show bs-rx-erroredPackets 1
bs-rx-errored packets for uiuc index 1 is 0
```

### bs-tx-amc-count-table

This table contains statistical information that can be used to characterize the adaptive modulation and coding performance in the downlink.

Command	Purpose	Privileges
FullMax(bs-private)#show bs-tx-amc-count-table <diuc-index>	Show all the parameters and their values of the TxAmcCountTable table for the given index	A/O

Example:

```
FullMax(bs-private)# show bs-tx-amc-count-table 1
bs-tx-octets for diuc index 1 is 0
bs-tx-packets for diuc index 1 is 0
```

### bs-tx-diuc-index

The Downlink Interval Usage Code indicates the downlink burst profile in the DCD message.

Command	Purpose	Privileges
FullMax(bs-private)#show bs-tx-diuc-index	Show all the available Downlink Interval Usage indices.	A/O

Example:

```
FullMax(bs-private)# show bs-tx-diuc-index
diuc indices are
```

```

bs-dl-diuc-index: 0
bs-dl-diuc-index: 1
bs-dl-diuc-index: 2
bs-dl-diuc-index: 3
bs-dl-diuc-index: 4
bs-dl-diuc-index: 5
bs-dl-diuc-index: 6
bs-dl-diuc-index: 7
    bs-dl-diuc-index: 8
bs-dl-diuc-index: 9
bs-dl-diuc-index: 10
bs-dl-diuc-index: 11
bs-dl-diuc-index: 12

```

### bs-tx-octets

This object counts the number of octets transmitted in the downlink using the downlink burst profile indexed by DiucIndex.

Command	Purpose	Privileges
FullMax(bs-private)# show bs-tx-octets <diuc-index>	Shows the number of octets transmitted in the downlink for the given index	A/O

Example:

```

FullMax(bs-private)# show bs-tx-octets 1
    bs-tx-octets for diuc index 1 is 0

```

### bs-tx-packets

This object counts the number of packets transmitted in the downlink using the downlink burst profile indexed by Diuc Index.

Command	Purpose	Privileges
FullMax(bs-rivate)#show bs-tx-packets <diuc-index>	Shows the number of packets transmitted in the downlink for the given index	A/O

Example:

```

FullMax(bs-private)# show bs-tx-packets 1
    bs-tx-packets for diuc index 1 is 0

```

### cmn-sf-table

This Table measures service flow traffic .

Command	Purpose	Privileges
FullMax(bs-private)#show cmn-sf-table <sfid>	This command show all the parameters and their values of Cmn-	A/O

	sf-table.	
--	-----------	--

Example:

```
FullMax(bs-private)# show cmn-sf-table 2001
sf-total-octets for sfid 2001 is 100
sf-total-pkts for sfid 2001 is 2
sf-errored-pkts for sfid 2001 is 1
sf-frag-orig for sfid 2001 is 1
sf-frag-total for sfid 2001 is 2
sf-missing-frag for sfid 2001 is 0
```

### cmn-pm-sfid

This is a 32 bit quantity that uniquely identifies a service flow to both the subscriber station and base station.

Command	Purpose	Privileges
FullMax(bs-private)#show cmn-pm-sfid <mac-ddress>	This command shows all the available Service Flow Id's Available for a particular MSID.	A/O

Example :

```
FullMax(bs-private)# show cmn-pm-sfid 112233:445566
cmn-pm-sfid 2001
cmn-pm-sfid 2002
cmn-pm-sfid 2003
cmn-pm-sfid 2006
```

### sf-total-octets

This determines the total octets received/transmitted on this service flow.

Command	Purpose	Privileges
FullMax(bs-private)# show sf-total-octets <sfid>	This command allows the user to display the total octets received or transmitted for this service-flow.	A/O

Example:

```
FullMax(bs-private)#show sf-total-octets 2001
sf-total-octets for sfid (2001) : 1000
```

### sf-total-pkts

This determines the total number of packets received /transmitted on this service flow.

Command	Purpose	Privileges
FullMax(bs-private)#show sf-total-pkts <sfid>	This Command allows the user to display the total packets received or trasmitted for this service-flow	A/O

Example:

```
FullMax(bs-private)#show sf-total-pkts 2001
sf-total-pkts for sfid (2001): 22
```

### sf-errored-pkts

This determines number of packets that were dropped due to missing fragments / bad ARQ blocks.

Command	Purpose	Privileges
FullMax(bs-private)#show sf-errored-pkts <sfid>	This Command Allows the user to display the total errored packets received or transmitted for this service flow.	A/O

Example:

```
FullMax(bs-private)# show sf-errored-pkts 2001
sf-errored-pkts for sfid (2001): 1
```

### sf-frag-orig

This determines number of fragments / ARQ blocks originally transmitted/received on this service flow.

Command	Purpose	Privileges
FullMax(bs-private)#show sf-frag-orig <sfid>	This Command allows the user to display the total fragments received or transmitted originally on this service flow.	A/O

Example:

```
FullMax(bs-private)#show sf-frag-orig 2001
sf-frag-orig for sfid (2001) : 2
```

### sf-frag-total

This determines total fragments / ARQ blocks transmitted / received on this service flow.

Command	Purpose	Privileges
FullMax(bs-private)#show sf-frag-total <sfid>	This Command allows the user to display the total fragments received or transmitted on this service flow.	A/O

Example:

```
FullMax(bs-private)# show sf-frag-total 2001
```

```
sf-frag-total for sfid (2001): 2
```

### sf-missing-frag

#### Description

This determines the number of missing received fragments received or NACK.

Command	Purpose	Privileges
FullMax(bs-private)# show sf-missing-frag <sfid>	This command allows the user to display the number of missing received fragments or NACK.	A/O

Example:

```
FullMax(bs-private)# show sf-missing-frag 2001
sf-missing-frag for sfid (2001): 1
```

## 5.2.15 trap-config

### trap-config

This Command allows the user to lock to trap config group.

Command	Purpose	Privileges
FullMax # trap- config	This command allows the user to Locks to the trap config group and displays the corresponding group prompt.	A/O

Example:

```
FullMax # trap-config
FullMax(trap-config) #
```

### std-trap-ctrl

The object is used to enable or disable Base Station traps From left to right, the set bit indicates the corresponding Base Station trap is enabled

```
{ wmanIf2BsSsStatusNotification (0),
  wmanIf2BsSsDynamicServiceFail (1),
  wmanIf2BsSsRssiStatusChange (2),
  wmanIf2BsSsRegister (3),
  wmanIf2BsSsPkmFail (4),
  wmanIf2BsPerformanceCounters (5)}
```

Command	Purpose	Privileges
FullMax(trap- config) # set std-trap-ctrl	This command Allows the user to update the value of std-trap-ctrl parameter.	A

<val>		
FullMax(trap-config) # show std-trap-ctrl	This command Allows the user to display the value of std-trap-ctrl parameter.	A/O

Example:

```
FullMax(trap-config) # set std-trap-ctrl 25
  updated bs-trap-control {wmanIf2BsSsStatusNotification(0),
                          wmanIf2BsSsRegister(3),
                          wmanIf2BsPerformanceCounters (4)
  }
```

```
FullMax(bs-private)# show std-trap-ctrl
bs-trap-control enabled are {
wmanIf2BsSsStatusNotification(0),
                          wmanIf2BsSsRegister(3),
                          wmanIf2BsPerformanceCounters
(4) }
```

**std-trap-status-ctrl**

The object is used to enable or disable Base Station status notification traps. The set bit indicates the corresponding Base Station trap is enabled.

```
{ unused(0), ssInitRangingSucc(1), ssInitRangingFail(2), ssRegistered(3),
  ssRegistrationFail(4), ssDeregistered(5), ssBasicCapabilitySucc(6),
  ssBasicCapabilityFail(7), ssAuthorizationSucc(8), ssAuthorizationFail(9),
  tftpSucc(10), tftpFail(11), sfCreationSucc(12), sfCreationFail(13) }
```

Command	Purpose	Privileges
FullMax(trap-config) # set std-trap-status-ctrl <val>	This command Allows the user to update the value of std-trap-status-ctrl parameter.	A
FullMax(trap-config) # show std-trap-status-ctrl	This command Allows the user to display the value of std-trap-status-ctrl parameter.	A/O

```
FullMax(bs-private)# set std-trap-status-ctrl 10900
  updated bs-trap-status-ctrl
                          {ssInitRangingFail(2),
ssRegistrationFail(4),
ssBasicCapabilityFail(7), ssAuthorizationFail(9),
                          tftpFail(11), sfCreationFail(13) }
```

```
FullMax(bs-private)# show std-trap-status-ctrl
bs-trap-status-ctrl enabled are
    {ssInitRangingFail(2),
ssRegistrationFail(4),
ssBasicCapabilityFail(7),ssAuthorizationFail(9),
    tftpFail(11), sfCreationFail(13) }
```

**private-trap-ctrl**

The object is used to enable or disable Base Station traps From left to right, the set bit indicates the corresponding Base Station trap is enabled.

```
{ wmanPriPowerStatusChange (0), wmanPriFanStatusChange (1),
wmanPriTemperatureChange (2), wmanPriConfigChange (3),
wmanPriAfeBitError (4) }
```

Command	Purpose	Privileges
FullMax(trap-config) # set private-trap-ctrl <val>	This command Allows the user to update the value of private-trap-ctrl parameter.	A
FullMax(trap-config) # show private-trap-ctrl	This command Allows the user to display the value of private-trap-ctrl parameter.	A/O

```
FullMax(bs-private)# set private-trap-ctrl 21
updated private-trap-ctrl
    { wmanPriPowerStatusChange(0),
wmanPriPowerStatusChange(2),
wmanPriPowerStatusChange (4) }
```

```
FullMax(bs-private)# show private-trap-ctrl
private-trap-ctrl enabled
    { wmanPriPowerStatusChange(0),
wmanPriPowerStatusChange(2),
wmanPriPowerStatusChange (4) }
```

**5.2.16 lapc-config**

**Lapc-config**

This group contains commands to manipulate the Link Adaption and Power Control configuration parameters for the Base Station MAC.

Command	Purpose
---------	---------

FullMax#lapc-config	The command allows the user to lock to config group and display the config prompt.
---------------------	--

```
FULLMAX#lapc-config
FULLMAX(lapc-config)]#
```

**uiuc**

Command	Purpose
FullMax(lapc-config)# show uiuc	Shows the values of uiuc available.

Example:

```
FULLMAX(lapc-config)#show uiuc
    uiuc 1
    uiuc 2
```

**Uplink\_FEC\_Code\_Type**

Command	Purpose
FullMax(lapc-config)# set Uplink_FEC_Code_Type <uiuc index> value	The command sets the value of Uplink_FEC_Code_Type for the given index
FullMax(lapc-config)# show Uplink_FEC_Code_Type <uiuc index>	The command show the value of Uplink_FEC_Code_Type for the particunlar uiuc index

Example:

```
FULLMAX(lapc-config)#set Uplink_FEC_Code_Type 1 333
    Updated Uplink_FEC_Code_Type for uiuc index 1 is 333
```

```
FULLMAX(lapc-config)#show Uplink_FEC_Code_Type 1
    Uplink_FEC_Code_Type for uiuc index 1 is 0
```

**Ranging\_Data\_Ratio**

Command	Purpose
FullMax(lapc-config)# set Ranging_Data_Ratio <uiuc index>	The command sets the value of Ranging_Data_Ratio for the given index.
FullMax(lapc-config)# show Ranging_Data_Ratio <uiuc index>	The command shows the value of Ranging_Data_Ratio for the given index.

Example:



```
FULLMAX(lapc-config)#set Ranging_Data_Ratio 1 22
Updated Ranging_Data_Ratio for uiuc index 1 is 22
```

```
FULLMAX(lapc-config)#show Ranging_Data_Ratio 1
Ranging_Data_Ratio for uiuc index 1 is 19
```

### Normalised\_C\_by\_N\_Mantissa

Command	Purpose
FullMax(lapc-config)# set Normalised_C_by_N_Mantis sa <uiuc index>	The commnad sets the value of Normalised_C_by_N_Mantissa for the given index
FullMax(lapc-config)# show Normalised_C_by_N_Mantis sa <uiuc index>	The commnad shows the value of Normalised_C_by_N_Mantissa for the given index

Example:

```
FULLMAX(lapc-config)#set Normalised_C_by_N_Mantissa 1 77
Updated Normalised_C_by_N_Mantissa for uiuc index 1 is
77
```

```
FULLMAX(lapc-config)#show Normalised_C_by_N_Mantissa 1
Normalised_C_by_N_Mantissa for uiuc index 1 is 60
```

### Normalised\_C\_by\_N\_Exponent

Command	Purpose
FullMax(lapc-config)# set Normalised_C_by_N_Expone nt <uiuc index>	The command sets the value of Normalised_C_by_N_Exponent for the given index
FullMax(lapc-config)# show Normalised_C_by_N_Expone nt <uiuc index>	The command shows the value of Normalised_C_by_N_Exponent for the given index

Example:

```
FULLMAX(lapc-config)#set Normalised_C_by_N_Exponent 1 1
Updated Normalised_C_by_N_Exponent for index 1 is 1
```

```
FULLMAX(lapc-config)#show Normalised_C_by_N_Exponent 1
Normalised_C_by_N_Exponent for index 1 is -1
```

### FIR\_filter\_coefficient\_mantissa

Command	Purpose
FullMax(lapc-config)# set FIR_filter_coefficient_mantissa <index> value	The command sets the value of FIR_filter_coefficient_mantissa for the given index
FullMax(lapc-config)# show FIR_filter_coefficient_mantissa <index>	The command shows the value of FIR_filter_coefficient_mantissa for the given index

Example:

```
FULLMAX(lapc-config)#set FIR_filter_coefficient_mantissa 1
65
    Updated FIR_filter_coefficient_mantissa for index 1 is
65
```

```
FULLMAX(lapc-config)#show FIR_filter_coefficient_mantissa 1
    FIR_filter_coefficient_mantissa for index 1 is 65
```

### **FIR\_filter\_coefficient\_TenPwr**

Command	Purpose
FullMax(lapc-config)#set FIR_filter_coefficient_TenPwr <index> <val>	The command sets the value of FIR_filter_coefficient_TenPwr for the given index.
FullMax(lapc-config)#show FIR_filter_coefficient_TenPwr <index>	The command shows the value of FIR_filter_coefficient_TenPwr for the given index.

Example:

```
FULLMAX(lapc-config)#set FIR_filter_coefficient_TenPwr 1 3
    Updated FIR_filter_coefficient_TenPwr for index 1 is 3
```

```
FULLMAX(lapc-config)#show FIR_filter_coefficient_TenPwr 1
    FIR_filter_coefficient_TenPwr for index 1 is 3
```

### **DI\_Benchmark\_Entry\_Threshold\_CINR**

Description

Command	Purpose
FullMax(lapc-config)#set DI_Benchmark_Entry_Threshold_CINR <dl_fec_code> <dl_repetition> <val>	Sets the value of DI_Benchmark_Entry_Threshold_CINR for the given fec code and repetition .
FullMax(lapc-config)#show DI_Benchmark_Entry_Threshold_CINR	Displays the value of DI_Benchmark_Entry_Threshold_CINR for the

INR <dl_fec_code> <dl_repetition>	given fec code and repetition .
--------------------------------------	---------------------------------

Example:

```
FULLMAX(lapc-config)#set Dl_Benchmark_Entry_Threshold_CINR
4 2 41e3
    Updated Dl_Benchmark_Entry_Threshold_CINR for fec code
4
    and repition 2 is 41e3
```

```
FULLMAX(lapc-config)#show Dl_Benchmark_Entry_Threshold_CINR
4 2
    Dl_Benchmark_Entry_Threshold_CINR for fec code 4 and
    repition 2 is 19e1
```

### **Dl\_Benchmark\_Exit\_Threshold\_CINR**

Description

<b>Command</b>	<b>Purpose</b>
FullMax(lapc-config)#set Dl_Benchmark_Exit_Threshold_CINR <dl_fec_code> <dl_repetition> <val>	Sets the value of Dl_Benchmark_Exit_Threshold_CINR for the given fec code and repetition .
FullMax(lapc-config)#show Dl_Benchmark_Exit_Threshold_CINR <dl_fec_code> <dl_repetition>	Displays the value of Dl_Benchmark_Exit_Threshold_CINR for the given fec code and repetition .

Example:

```
FULLMAX(lapc-config)#set Dl_Benchmark_Exit_Threshold_CINR 4
2 41e3
    Updated Dl_Benchmark_Exit_Threshold_CINR for fec code
4 and
    repition 2 is 41e3
```

```
FULLMAX(lapc-config)#show Dl_Benchmark_Exit_Threshold_CINR
4 2
    Dl_Benchmark_Exit_Threshold_CINR for fec code 4 and
    repition 2 is 19e1
```

### **Ul\_Benchmark\_Entry\_Threshold\_CINR**

<b>Command</b>	<b>Purpose</b>
FullMax(lapc-config)#set Ul_Benchmark_Entry_Threshold_CINR <dl_fec_code> <dl_repetition> <val>	Sets the value of Ul_Benchmark_Entry_Threshold_CINR for the given fec code and repetition .

INR <dl_fec_code> <dl_repetition> <val>	given fec code and repetition .
FullMax(lapc-config)#show Ul_Benchmark_Entry_Threshold_C INR <dl_fec_code> <dl_repetition>	Displays the value of Ul_Benchmark_Entry_Threshold_CINR for the given fec code and repetition .

```
FULLMAX(lapc-config)#set Ul_Benchmark_Entry_Threshold_CINR
4 2 41e3
Updated Ul_Benchmark_Entry_Threshold_CINR for fec code
4
and repition 2 is 41e3
```

```
FULLMAX(lapc-config)#show Ul_Benchmark_Entry_Threshold_CINR
4 2
Ul_Benchmark_Entry_Threshold_CINR for fec code 4 and
repetition 2 is 19e1
```

### **UI\_Benchmark\_Exit\_Threshold\_CINR**

Description

<b>Command</b>	<b>Purpose</b>
FullMax(lapc-config)#set Ul_Benchmark_Exit_Threshold_CI NR <ul_fec_code> <ul_repetition> <val>	Sets the value of Ul_Benchmark_Exit_Threshold_CINR for the given fec code and repetition .
FullMax(lapc-config)#show Ul_Benchmark_Exit_Threshold_CI NR <ul_fec_code> <ul_repetition>	Displays the value of Ul_Benchmark_Exit_Threshold_CINR for the given fec code and repetition .

Example:

```
FULLMAX(lapc-config)#set Dl_Benchmark_Exit_Threshold_CINR 4
2 41e3
Updated Dl_Benchmark_Exit_Threshold_CINR for fec code
4 and repition 2 is 41e3
```

```
FULLMAX(lapc-config)#show Dl_Benchmark_Exit_Threshold_CINR
4 2
Dl_Benchmark_Exit_Threshold_CINR for fec code 4 and
repetition 2 is 19e1
```

### **Burst\_Profile\_evaluation\_Timer**

<b>Command</b>	<b>Purpose</b>
----------------	----------------

FullMax(lapc-config)#set Burst_Profile_evaluation_Timer	The command sets the value of _Profile_evaluation_Timer
FullMax(lapc-config)#show Burst_Profile_evaluation_Timer	The command shows the value of _Profile_evaluation_Timer

Example:

```
FULLMAX(lapc-config)#set Burst_Profile_evaluation_Timer 222
Updated Burst_Profile_evaluation_Timer: 250
```

```
FULLMAX(lapc-config)#show Burst_Profile_evaluation_Timer
Burst_Profile_evaluation_Timer: 250
```

### **Mantissa\_Lower\_Thld\_Rx\_Power\_Density**

Description

<b>Command</b>	<b>Purpose</b>
FullMax(lapc-config)# set Mantissa_Lower_Thld_Rx_Power_Density <val>	Sets the value of Mantissa_Lower_Thld_Rx_Power_Density
FullMax(lapc-config)# show Mantissa_Lower_Thld_Rx_Power_Density	Displays the value of Mantissa_Lower_Thld_Rx_Power_Density.

Example:

```
FULLMAX(lapc-config)#set
Mantissa_Lower_Thld_Rx_Power_Density 22
Updated Mantissa_Lower_Thld_Rx_Power_Density -100
```

```
FULLMAX(lapc-config)#show
Mantissa_Lower_Thld_Rx_Power_Density
Mantissa_Lower_Thld_Rx_Power_Density -100
```

### **Mantissa\_Higher\_Thld\_Rx\_Power\_Density**

Description

<b>Command</b>	<b>Purpose</b>
FullMax(lapc-config)#set Mantissa_Higher_Thld_Rx_Power_Density	Sets the value of Mantissa_Higher_Thld_Rx_Power_Density to the given value.
FullMax(lapc-config)#show Mantissa_Higher_Thld_Rx_Power_Density	Display the value of Mantissa_Higher_Thld_Rx_Power_Density

Example:

```
FULLMAX(lapc-config)#set
Mantissa_Higher_Thld_Rx_Power_Density 22
Updated Mantissa_Higher_Thld_Rx_Power_Density 22

FULLMAX(lapc-config)#show
Mantissa_Higher_Thld_Rx_Power_Density
Mantissa_Higher_Thld_Rx_Power_Density -60
```

### Max\_Length\_Of\_DLMAP\_Mesg

Command	Purpose
FullMax(lapc-config)#set Max_Length_Of_DLMAP_Mesg <value>	The command sets the value of Length_Of_DLMAP_Mesg to the value given
FullMax(lapc-config)#show Max_Length_Of_DLMAP_Mesg	The command shows the value of Length_Of_DLMAP_Mesg

Example:

```
FULLMAX(lapc-config)#set Max_Length_Of_DLMAP_Mesg 22
Updated Max_Length_Of_DLMAP_Mesg: 22
```

```
FULLMAX(lapc-config)#show Max_Length_Of_DLMAP_Mesg
Max_Length_Of_DLMAP_Mesg: 2
```

### Max\_Aggregate\_Length\_Of\_Control\_Messages

Command	Purpose
FullMax(lapc-config)# set Max_Aggregate_Length_Of_Control_M essages <value>	The command sets the value of Aggregate_Length_Of_Control_Messages to the given value.
FullMax(lapc-config)# show Max_Aggregate_Length_Of_Control_M essages	The command shows the value of Aggregate_Length_Of_Control_Messages.

Example:

```
FULLMAX(lapc-config)#set
Max_Aggregate_Length_Of_Control_Messages 22
Updated Max_Aggregate_Length_Of_Control_Messages: 22
```

```
FULLMAX(lapc-config)#show
Max_Aggregate_Length_Of_Control_Messages
Max_Length_Of_DLMAP_Mesg: 48
```

### Max\_No\_Of\_Slots\_Used\_For\_Future\_Allocation\_In\_DL

<b>Command</b>	<b>Purpose</b>
FullMax(lapc-config)# set Max_No_Of_Slots_Used_For_Future_Allocation_In_DL <value>	The command sets the value of No_Of_Slots_Used_For_Future_Allocation_In_DL to the given value.
FullMax(lapc-config)# show Max_No_Of_Slots_Used_For_Future_Allocation_In_DL	The command shows the value of No_Of_Slots_Used_For_Future_Allocation_In_DL.

Example:

```
[FULLMAX(lapc-config)]$set
Max_No_Of_Slots_Used_For_Future_Allocation_In_DL 33
```

```
Updated Max_No_Of_Slots_Used_For_Future_Allocation_In_DL:
33
```

```
[FULLMAX(lapc-config)]$show
Max_No_Of_Slots_Used_For_Future_Allocation_In_DL
```

```
Max_Aggregate_Length_Of_Control_Messages: 90
```

### **Max\_No\_Of\_Slots\_Used\_For\_Future\_Allocation\_In\_UL**

<b>Command</b>	<b>Purpose</b>
FullMax(lapc-config)# set Max_No_Of_Slots_Used_For_Future_Allocation_In_UL <value>	The command sets the value of No_Of_Slots_Used_For_Future_Allocation_In_UL to the given value.
FullMax(lapc-config)# show Max_No_Of_Slots_Used_For_Future_Allocation_In_UL	The command shows the value of No_Of_Slots_Used_For_Future_Allocation_In_UL.

Example:

```
FULLMAX(lapc-config)#set
Max_No_Of_Slots_Used_For_Future_Allocation_In_UL 44
```

```
Updated Max_No_Of_Slots_Used_For_Future_Allocation_In_DL:
44
```

```
FULLMAX(lapc-config)#show
Max_No_Of_Slots_Used_For_Future_Allocation_In_UL
```

```
Max_No_Of_Slots_Used_For_Future_Allocation_In_DL: 10
```

### **Max\_No\_Of\_Slots\_Can\_Be\_Used\_For\_CDMA\_Allocation\_In\_Single\_UL\_Frame**

Command	Purpose
FullMax(lapc-config)#set Max_No_Of_Slots_Can_Be_Used_For_CDMA_Allocation_In_Single_UL_Frame <value>	The command sets the value of No_Of_Slots_Can_Be_Used_For_CDMA_Allocation_In_Single_UL_Frame to the given value.
FullMax(lapc-config)#show Max_No_Of_Slots_Can_Be_Used_For_CDMA_Allocation_In_Single_UL_Frame	The command shows the value of No_Of_Slots_Can_Be_Used_For_CDMA_Allocation_In_Single_UL_Frame.

Example:

```
FULLMAX(lapc-config)#set  
Max_No_Of_Slots_Can_Be_Used_For_CDMA_Allocation_In_Single_UL_Frame 22
```

Updated

```
Max_No_Of_Slots_Can_Be_Used_For_CDMA_Allocation_In_Single_UL_Frame: 22
```

```
FULLMAX(lapc-config)#show
```

```
Max_No_Of_Slots_Can_Be_Used_For_CDMA_Allocation_In_Single_UL_Frame
```

```
Max_No_Of_Slots_Used_For_Future_Allocation_In_DL: 10
```

### **No\_Of\_CDMA\_Ranging\_Slots**

Command	Purpose
FullMax(lapc-config)# set No_Of_CDMA_Ranging_Slots <value>	The command sets the value of No_Of_CDMA_Ranging_Slots to the given value.
FullMax(lapc-config)# show No_Of_CDMA_Ranging_Slots	The command shows the value of No_Of_CDMA_Ranging_Slots.

```
FULLMAX(lapc-config)#set No_Of_CDMA_Ranging_Slots 33
```

```
Updated No_Of_CDMA_Ranging_Slots: 33
```

```
FULLMAX(lapc-config)#show No_Of_CDMA_Ranging_Slots
```

```
No_Of_CDMA_Ranging_Slots: 1
```

### **CDMA\_Ranging\_Period**

Command	Purpose
FullMax(lapc-config)# set CDMA_Band_Width_Period <value>	The command sets the value of CDMA_Band_Width_Period to the given value.
FullMax(lapc-config)# show	The command shows the value of



CDMA_Band_Width_Period	CDMA_Band_Width_Period
------------------------	------------------------

```
FULLMAX(lapc-config)#set CDMA_Band_Width_Period 33
Updated CDMA_Band_Width_Period: 33
```

```
FULLMAX(lapc-config)#show CDMA_Band_Width_Period
CDMA_Band_Width_Period: 6
```

### No\_Of\_CDMA\_Band\_Width\_Slots

Command	Purpose
FullMax(lapc-config)# set No_Of_CDMA_Band_Width_Slots <value>	The command sets the value of No_Of_CDMA_Band_Width_Slots to the given value
FullMax(lapc-config)# show No_Of_CDMA_Band_Width_Slots	The command shows the value of No_Of_CDMA_Band_Width_Slots

Example:

```
FULLMAX(lapc-config)#set No_Of_CDMA_Band_Width_Slots 33
Updated No_Of_CDMA_Band_Width_Slots: 33
```

```
FULLMAX(lapc-config)#show No_Of_CDMA_Band_Width_Slots
No_Of_CDMA_Band_Width_Slots: 3
```

### CDMA\_Band\_Width\_Period

Command	Purpose
FullMax(lapc-config)# set CDMA_Band_Width_Period <value>	The command sets the value of CDMA_Band_Width_Period to the given value.
FullMax(lapc-config)# show CDMA_Band_Width_Period	The command shows the value of CDMA_Band_Width_Period.

Example:

```
FULLMAX(lapc-config)#set CDMA_Band_Width_Period 33
Updated CDMA_Band_Width_Period: 33
```

```
FULLMAX(lapc-config)#show CDMA_Band_Width_Period
CDMA_Band_Width_Period: 6
```

### Max\_Frame\_Slots\_Dl\_Harq\_Retransmission

Command	Purpose
FullMax(lapc-config)# set Max_Frame_Slots_Dl_Harq_Retransmission <value>	The command sets the value of Max_Frame_Slots_Dl_Harq_Retransmission to the given value.

FullMax(lapc-config)# show Max_Frame_Slots_Dl_Harq_Retransmission	The command shows the value of _Frame_Slots_Dl_Harq_Retransmission.
--	--

Example:

```
FULLMAX(lapc-config)#set
Max_Frame_Slots_Dl_Harq_Retransmission 33
Updated Max_Frame_Slots_Dl_Harq_Retransmission 33
```

```
FULLMAX(lapc-config)#show
Max_Frame_Slots_Dl_Harq_Retransmission
Max_Frame_Slots_Dl_Harq_Retransmission: 20
```

### Max\_Frame\_Slots\_UI\_Harq\_Retransmission

Command	Purpose
FullMax(lapc-config)# set Max_Frame_Slots_UI_Harq_Retransmission <value>	The command sets the value of Max_Frame_Slots_UI_Harq_Retransmission to the given value.
FullMax(lapc-config)# show Max_Frame_Slots_UI_Harq_Retransmission	The command shows the value of _Frame_Slots_UI_Harq_Retransmission.

Example:

```
FULLMAX(lapc-config)#set
Max_Frame_Slots_UI_Harq_Retransmission 33
Update Max_Frame_Slots_UI_Harq_Retransmission: 33
```

```
FULLMAX(lapc-config)#show
Max_Frame_Slots_UI_Harq_Retransmission
Max_Frame_Slots_UI_Harq_Retransmission: 20
```

### Max\_UI\_Harq\_Ack\_slots

Command	Purpose
FullMax(lapc-config)# set Max_UI_Harq_Ack_slots <value>	The command sets the value of Max_UI_Harq_Ack_slots to the givan value.
FullMax(lapc-config)# show Max_UI_Harq_Ack_slots	The command shows the value of Max_UI_Harq_Ack_slots

Example:

```
FULLMAX(lapc-config)#set Max_UI_Harq_Ack_slots 24
Updated Max_UI_Harq_Ack_slots: 24
```

```
FULLMAX(lapc-config)#show Max_UI_Harq_Ack_slots
```

Max\_Ul\_Harq\_Ack\_slots: 2

### Initial\_Ranging\_Backoff\_Start

Command	Purpose
FullMax(lapc-config)# set Initial_Ranging_Backoff_Start <value>	The command sets the value of <u>Initial_Ranging_Backoff_Start</u> to the given value.
FullMax(lapc-config)# show Initial_Ranging_Backoff_Start	The command shows the value of <u>Initial_Ranging_Backoff_Start</u>

Example:

```
FULLMAX(lapc-config)#set Initial_Ranging_Backoff_Start
55
Updated Initial_Ranging_Backoff_Start: 55
```

```
FULLMAX(lapc-config)#show Initial_Ranging_Backoff_Start
Initial_Ranging_Backoff_Start: 15
```

### Initial\_Ranging\_Backoff\_End

Command	Purpose
FullMax(lapc-config)# set Initial_Ranging_Backoff_End <value>	The command sets the value of <u>Initial_Ranging_Backoff_End</u> to the given value.
FullMax(lapc-config)# show Initial_Ranging_Backoff_End	The command shows the value of <u>Initial_Ranging_Backoff_End</u> .

```
FULLMAX(lapc-config)#set Initial_Ranging_Backoff_End 44
Updated Initial_Ranging_Backoff_End: 44
```

```
FULLMAX(lapc-config)#show Initial_Ranging_Backoff_End
Initial_Ranging_Backoff_End: 12
```

### Bandwidth\_Request\_Backoff\_Start

Command	Purpose
FullMax(lapc-config)# set Bandwidth_Request_Backoff_Start <value>	The command sets the value of <u>Bandwidth_Request_Backoff_Start</u> to the given value.
FullMax(lapc-config)# show Bandwidth_Request_Backoff_Start	The command shows the value of <u>Bandwidth_Request_Backoff_Start</u> .

Example:

```
FULLMAX(lapc-config)#set Bandwidth_Request_Backoff_Start 44
Updated Bandwidth_Request_Backoff_Start: 44
```

```
FULLMAX(lapc-config)#show Bandwidth_Request_Backoff_Start
Bandwidth_Request_Backoff_Start: 3
```

### Bandwidth\_Request\_Backoff\_End

Command	Purpose
FullMax(lapc-config)# set Bandwidth_Request_Backoff_End <value>	The command sets the value of Request_Backoff_End to the given value.
FullMax(lapc-config)# show Bandwidth_Request_Backoff_End	The command shows the value of Request_Backoff_End.

Example:

```
FULLMAX(lapc-config)#set Bandwidth_Request_Backoff_End 55
Updated Bandwidth_Request_Backoff_End: 55
```

```
FULLMAX(lapc-config)#show Bandwidth_Request_Backoff_End
Bandwidth_Request_Backoff_End: 2
```

### Percentage\_Of\_Reserved\_Symbols

Command	Purpose
FullMax(lapc-config)# set Percentage_Of_Reserved_Symbols <value>	The command sets the value of Of_Reserved_Symbols to the given value.
FullMax(lapc-config)# show Percentage_Of_Reserved_Symbols	The command shows the value of Of_Reserved_Symbols.

Example:

```
FULLMAX(lapc-config)#set Percentage_Of_Reserved_Symbols 99
Updated Percentage_Of_Reserved_Symbols: 99
```

```
FULLMAX(lapc-config)#show Percentage_Of_Reserved_Symbols
Percentage_Of_Reserved_Symbols: 10
```

### Percentags\_Of\_Head\_Room\_For\_MAC

Command	Purpose
FullMax(lapc-config)# set Percentags_Of_Head_Room_For_MAC <value>	The command sets the value of Of_Head_Room_For_MAC to the given value.

FullMax(lapc-config)# show Percentags_Of_Head_Room_For_MAC	The command shows the value of Of_Head_Room_For_MAC.
--	--

Example:

```
FULLMAX(lapc-config)#set Percentags_Of_Head_Room_For_MAC 88
Updated Percentags_Of_Head_Room_For_MAC: 88
```

```
FULLMAX(lapc-config)#show Percentags_Of_Head_Room_For_MAC
Percentags_Of_Head_Room_For_MAC: 10
```

### 5.3 MS400 Operation

The FullMax MS CLI supports the following commands. The access to this commands are based on the user privileges. The Admin has the privilege to control and monitor all the information supported by the FullMax MS CLI commands, where as the operator has limited privileges.

After the successful authentication, the user gets the access permission to the command line interface. Based on the user privileges, the help menu will be displayed to the user.

#### 5.3.1 Main Group

When user logs in to the CLI it will lock to the default group which is the main group. When ever the user enters the Main group, CLI will display the main prompt. Help command in main group will show the commands supported in main group along with the group lock commands. Only commands related to main group can be executed in main group.

```
FullMax# main group
```

#### ip-address

System IP address.

Command	Purpose	Privileges
FullMax# set ip-address <ip-address>	Updates the system IP address.	A/O
FullMax# show ip-address	Displays the system IP address.	A/O

Example:

```
FullMax#set ip-address 10.60.4.42
```

```
Updated IP address: 10.60.4.42
```

```
FullMax# show ip-address
```

```
IP address: 10.60.4.56
```

### 5.3.2 ss-config

The ms-config group define the commands that monitors or updates the configuration information. The user upon entering the ms-config command locks into the CLI to execute commands only related to MS configuration.

```
FullMax#
FullMax# ss-config
FullMax(ss-config)#
```

Command	Purpose	Privileges
FullMax# ss-config	The command allows the user to lock into the group ss-config and execute all the commands related to the group.	A/O

The commands supported by the ss-config group are.

#### Lost-dl-map-interval

Time since last received DL-MAP message before downlink synchronization is considered lost in ms.

Command	Purpose	Privileges
FullMax(ss-config)# set lost-dl-map-interval <val>	The command allows the user to update timer value since last received DL-MAP message before downlink synchronization is considered lost.	A
FullMax(ss-config)# show lost-dl-map-interval	The command allows the user to display the timer value since last received DL-MAP message before downlink synchronization is considered lost.	A/O

Example:

```
FullMax(ss-config) # set lost-dl-map-interval 300
Updated lost-dl-map-interval 300 milliseconds
```

```
FullMax (ss-config) # show lost-dl-map-interval
lost-dl-map-interval 300 milliseconds
```

#### Lost-ul-map-interval

Time since last received UL-MAP message before downlink synchronization is considered lost in ms.

Command	Purpose	Privileges
---------	---------	------------

FullMax(ss-config)# set lost-ul-map-interval <val>	The command allows the user to update timer value since last received UL-MAP message before downlink synchronization is considered lost.	A
FullMax(ss-config)# show lost-ul-map-interval	The command allows the user to display the timer value since last received UL-MAP message before downlink synchronization is considered lost.	A/O

Example:

```
FullMax(ss-config)# set lost-ul-map-interval 300
Updated lost-ul-map-interval 300 milliseconds
```

```
FullMax(ss-config)# show lost-ul-map-interval
lost-ul-map-interval 300 milliseconds
```

### Contention-rng-retries

Number of retries on contention ranging requests.

Command	Purpose	Privileges
FullMax(ss-config)# set contention-rng-retries <val>	The command allows the user to update the contention ranging retries parameter.	A
FullMax(ss-config)# show contention-rng-retries	The command allows the user to display the contention ranging retries parameter.	A/O

Example:

```
FullMax(ss-config)# set contention-rng-retries 20
Updated contention-rng-retries 20
```

```
FullMax(ss-config)# show contention-rng-retries
contention-rng-retries 20
```

### Request-retries

Number of retries on bandwidth allocation request.

Command	Purpose	Privileges
FullMax(ss-config)# set request-retries <val>	The command allows the user to update the number of retries on bandwidth allocation request.	A
FullMax(ss-config)# show request-retries	The command allows the user to display the number of retries on bandwidth allocation request.	A/O

Example:

```
FullMax(ss-config)# set request-retries 88
Updated request-retries 88
```

```
FullMax(ss-config)# show request-retries
request-retries 88
```

### Reg-requeset-retries

Number of retries on registration request.

Command	Purpose	Privileges
FullMax(ss-config)# set reg-requeset-retries <val>	The command allows the user to update the number of retries on registration request.	A
FullMax(ss-config)# show reg-requeset-retries	The command allows the user to display the number of retries on registration request.	A/O

Example:

```
FullMax(ss-config)# set reg-requeset-retries 8
Updated reg-requeset-retries 8
```

```
FullMax(ss-config)# show reg-requeset-retries
reg-requeset-retries 8
```

### T1-timeout

Wait for DCD timeout in milliseconds.

Command	Purpose	Privileges
FullMax(ss-config)# set t1-timeout <val>	The command allows the user to update the DCD timeout parameter.	A
FullMax(ss-config)# show t1-timeout	The command allows the user to display the DCD timeout parameter.	A/O

Example:

```
FullMax(ss-config)# set t1-timeout 9000
Updated t1-timeout 9000 milliseconds.
```

```
FullMax(ss-config)# show t1-timeout
t1-timeout 9000 milliseconds.
```

### T2-timeout

Wait for broadcast ranging timeout in milliseconds.



Command	Purpose	Privileges
FullMax(ss-config)# set t2-timeout <val>	The command allows the user to update the broadcast ranging timeout parameter	A
FullMax(ss-config)# show t2-timeout	The command allows the user to display the broadcast ranging timeout parameter	A/O

Example:

```
FullMax(ss-config) # set t2-timeout 8000
Updated t2-timeout 8000 milliseconds.
```

```
FullMax(ss-config)# show t2-timeout
t2-timeout 8000 milliseconds.
```

### T3-timeout

Ranging response reception timeout following the transmission of Ranging Request in milliseconds.

Command	Purpose	Privileges
FullMax(ss-config)# set t3-timeout <val>	The command allows the user to update the ranging response timeout parameter.	A
FullMax(ss-config)# show t3-timeout	The command allows the user to display the ranging response timeout parameter.	A/O

Example:

```
FullMax(ss-config)# set t3-timeout 45
Updated t3-timeout 45 milliseconds.
```

```
FullMax(ss-config)# show t3-timeout
T3-timeout 45 milliseconds.
```

### T4-timeout

Wait for ranging opportunity or data grant. If pending until complete field was used earlier by this SS, then the value of that field shall be added to this interval in second.

Command	Purpose	Privileges
FullMax(ss-config)# set t4-timeout <val>	The command allows the user to update the wait for ranging opportunity timeout parameter.	A
FullMax(ss-config)# show t4-timeout	The command allows the user to display the wait for ranging opportunity timeout parameter.	A/O

Example:

```
FullMax(ss-config) # set t4-timeout 45
Updated t4-timeout 45 milliseconds.
```

```
FullMax(ss-config)# show t4-timeout
T4-timeout 45 milliseconds.
```

### T16-timeout

Wait for Registration Response in milliseconds.

Command	Purpose	Privileges
FullMax(ss-config)#set t16-timeout <val>	The command allows the user to update the Registration response wait timer.	A
FullMax (ss-config)#show t16-timeout	The command allows the user to display the Registration response wait timer.	A/O

Example:

```
FullMax(ss-config)#set t16-timeout 540
Updated t16-timeout 540 milliseconds.
```

```
FullMax (ss-config) # show t16-timeout
T16-timeout 540 milliseconds.
```

### T12-timeout

Wait for UCD descriptor in milliseconds.

Command	Purpose	Privileges
FullMax(ss-config)# set t12-timeout <val>	The command allows the user to update wait for UCD descriptor timer.	A
FullMax(ss-config)# show t12-timeout	The command allows the user to display the wait for UCD descriptor timer.	A/O

Example:

```
FullMax(ss-config) # set t12-timeout 12000
Updated t12-timeout 12000 milliseconds.
```

```
FullMax (ss-config) # show t12-timeout
t12-timeout 12000 milliseconds.
```

### T18-timeout

Wait for SBC-RSP timeout in milliseconds.

Command	Purpose	Privileges
FullMax(ss-config)# set t18-timeout <val>	The command allows the user to update wait for SBC-RSP timeout parameter.	A
FullMax(ss-config)# show t18-timeout	The command allows the user to display the wait for SBC-RSP timeout parameter.	A/O

Example:

```
FullMax(ss-config) # set t18-timeout 7000
Updated t18-timeout 7000 milliseconds.
```

```
FullMax (ss-config) # show t18-timeout
t18-timeout 7000 milliseconds.
```

### T19-timeout

Time DL-channel remains unusable in ms.

Command	Purpose	Privileges
FullMax(ss-config)#set t19-timeout <val>	The command allows the user to update time for DL-channel remains unusable in ms..	A
FullMax(ss-config)#show t19-timeout	The command allows the user to display the time for DL-channel remains unusable in ms.	A/O

Example:

```
FullMax(ss-config)# set t19-timeout 7000
Updated t19-timeout 7000 milliseconds.
```

```
FullMax(ss-config)# show t19-timeout
t19-timeout 7000 milliseconds.
```

### T20-timeout

Time SS searches for preambles on a given channel in milliseconds.

Command	Purpose	Privileges
FullMax(ss-config)#set t20-timeout <val>	The command allows the user to update the time to search for preambles on a given channel.	A
FullMax(ss-config)#show t20-timeout	The command allows the user to display time to search for preambles on a given channel.	A/O

Example:

```
FullMax(ss-config) # set t20-timeout 600
    Updated t20-timeout 600 milliseconds.
```

```
FullMax (ss-config) # show t20-timeout
    t20-timeout 600 milliseconds.
```

### T21-timeout

Time SS searches for DL-MAP on a given channel in milliseconds.

Command	Purpose	Privileges
FullMax(ss-config) #set t21-timeout <val>	The command allows the user to update the time to search for DL-MAP on a given channel.	A
FullMax(ss-config) #show t21-timeout	The command allows the user to display time to search for DL-MAP on a given channel.	A/O

Example:

```
FullMax(ss-config)# set t21-timeout 1200
    Updated t21-timeout 1200 milliseconds.
```

```
FullMax(ss-config)# show t21-timeout
    t21-timeout 1200 milliseconds.
```

### 5.3.3 device

The device group define the commands that monitors or updates the device related information in the private MIB. The user upon entering the device command locks into the CLI to execute commands only related to device.

```
FullMax#
    FullMax# device
FullMax(device)#
```

Command	Purpose	Privileges
FullMax#	The command allows the user to lock into the group device and execute all the commands related to the group.	A/O

### type

Type of the device (BS / MS / SS).

Command	Purpose	Privileges
FullMax(device)#show type	The command allows the user display the device type (BS/MS/SS)	A/O

Example:

```
FullMax(device)# show type
type SS
```

### gpos

The geographical position of the device ,  
i.e. the real number describing the longitude and latitudes encoded as a printable string.  
Longitude - the precision is within the range -90..90 degrees. Positive numbers indicate locations north of the equator.  
Latitude - The precision is within the range -180..180 degrees. Positive numbers indicate locations east of the prime meridian

Command	Purpose	Privileges
FullMax(device)#set gpos <val>	The command allows the user to send the command to the GPS available on the system.	A
FullMax(device)#show gpos	The command allows the user display geographical position of the device.	A/O

Example:

```
FullMax(device)# set gpos 64.000 88.3400
updated gpos 64.000 88.3400
FullMax(device)# show gpos
gpos 64.000 88.3400
```

### boot-time

The absolute time of last device boot up.

Command	Purpose	Privileges
FullMax(device)# show boot-	timeThe command allows the user display the device boot up time.	A/O

Example:

```
FullMax(device)# show boot-time
boot-time 230404:18052008
```

### commit-save

Setting this object to TRUE causes the device to write all configuration changes in FLASH memory. On next boot the changes will be relevant. If this operation does not occur, configuration changes will not be maintained through reset Reading this object always returns FALSE

Command	Purpose	Privileges
FullMax(device)#set commit-save <val>	The command allows the user to set the value to TRUE so as to save the configuration changes in FLASH.	A
FullMax(device)#show commit-save	The command allows the user display the device commit-save	A/O

Example:

```
FullMax(device)# set commit-save
commit-save TRUE
```

```
FullMax(device)# show commit-save
commit-save FALSE
```

### gps-card

GPS card availability in the device.

Command	Purpose	Privileges
FullMax(device)#show gps-card	The command allows the user display the availability of GPS card in the device.	A/O

Example:

```
FullMax(device)# show gps-card
gps-card YES
```

### bpc-hw-version

Hardware version of baseband processor card.

Command	Purpose	Privileges
FullMax(device)#show bpc-hw-version	The command allows the user display the hardware version of baseband processor card.	A/O

Example:

```
FullMax(device)# show bpc-hw-version
bpc-hw-version 1.10
```

### afe-hw-ver

Hardware version of the Analog Front End (AFE)

Command	Purpose	Privileges
FullMax(device)#show afe-hw-ver	The command allows the user display the hardware version of AFE.	A/O

--	--	--

Example:

```
FullMax(device)# show afe-hw-ver
device-afe-hw-version 1.3
```

**afe-sw-ver**

Software version of the Analog Front End (AFE)

Command	Purpose	Privileges
FullMax(device)#show afe-sw-ver	The command allows the user display the software version of AFE.	A/O

Example:

```
FullMax(device)# show afe-sw-ver
device-afe-sw-version 1.3
```

**build-sw-ver**

Software version of the device general embedded software.

Command	Purpose	Privileges
FullMax(device)#show build-sw-ver	The command allows the user display the software version of the software build	A/O

Example:

```
FullMax(device)#show build-sw-ver
build-sw-version
```

**5.3.4 Measurements**

The Measurement group define the commands that monitors or updates the measurement related information in the private MIB. The user upon entering the device command locks into the CLI to execute commands only related to Measurement .

```
FullMax#
FullMax# measurement
FullMax(measurement)#
```

Command	Purpose	Privileges
FullMax# measurement	The command allows the user to lock into the group Measurement and execute all the commands related to this group.	A/O

**Temperature**

Temperature degree in Celsius

Command	Purpose	Privileges
FullMax(measurement)# show temperature	The command allows the user display temperature of the device.	A/O
FullMax(measurement)# track [-rN] [-iN] temperature	The command allows the user to display real time temperature sample of the device.	A/O

Example:

```
FullMax(measurement)# show temperature
    Temperature 45 degrees-Celsius
FullMax(measurement)# track -r3 -i2 temperature
    Temperature 45 degrees-Celsius
    Temperature 46 degrees-Celsius
    Temperature 45 degrees-Celsius
```

**Voltage**

Voltage measurement

Command	Purpose	Privileges
FullMax(measurement)# show voltage	The command allows the user display voltage of the device.	A/O
FullMax(measurement)# track [-rN] [-iN] voltage	The command allows the user to display multiple voltage samples of the device.	A/O

Example:

```
FullMax(measurement)# show voltage
    Voltage 3 volts
FullMax(measurement)# track -r3 -i10 voltage
    Voltage 3 volts
    Voltage 3 volts
    Voltage 3 volts
```

**Current**

Current measurements

Command	Purpose	Privileges
FullMax(measurement)#show current	The command allows the user display the current in the device.	A/O
FullMax(measurement)#track	The command allows the user to	A/O



<code>[-rN] [-iN] current.</code>	display multiple current measurements in the device	
-----------------------------------	---	--

Example:

```
FullMax(measurement)# show current
    Current 2 amp
FullMax(measurement)# track -r4 current
    Current 2 amp
    Current 2 amp
    Current 2 amp
```

### Tx-power

Transmit power

Command	Purpose	Privileges
<code>FullMax(measurement)#show tx-power</code>	The command allows the user display the transmit power.	A/O
<code>FullMax(measurement #track [-rN] [-iN] tx-power</code>	The command allows the user to display multiple transmit power measurements.	A/O

Example:

```
FullMax(measurement)# show tx-power
    Tx-power 45 dBm
FullMax(measurement)# track -r2 tx-power
    Tx-power 45 dBm
    Tx-power 43 dBm
```

### Rx-power

Receive power

Command	Purpose	Privileges
<code>FullMax(measurement)#show rx-power</code>	The command allows the user display the receive power.	A/O
<code>FullMax(measurement)#track [-rN] [-iN] rx-power</code>	The command allows the user to display multiple receive power measurements.	A/O

Example:

```
FullMax(measurement)# show rx-power
    rx-power 45 dBm
FullMax(measurement)# track -r2 -i2 rx-power
    rx-power 45 dBm
    rx-power 45 dBm
```

**afe-temperature**

Read the temperature from the Analog Front End.

Command	Purpose	Privileges
FullMax(measurement)#show afe-temperature	The command allows the user display temperatue from Analog Front End.	A/O.
FullMax(measurement)#track [-rN] [-iN] afe-temperature	The command allows the user to display multiple temperatue measurements from Analog Front End	A/O

Example:

```
FullMax(measurement)# show afe-temperatue
afe-temperature 27 celcius
FullMax(measurement)# track -r4 -i30 afe-temperatue
afe-temperature 27 celcius
afe-temperature 27 celcius
afe-temperature 27 celcius
afe-temperature 27 celcius
```

**afe-rssi**

Read the Receive Signal Strength Indicator (RSSI) from the Analog Front End.

Command	Purpose	Privileges
FullMax(measurement)#show afe-rssi	The command allows the user display RSSI from Analog Front End.	A/O
FullMax(measurement)#track [-rN] [-iN] afe-rssi	The command allows the user to display multiple RSSI measurements from Analog Front End .	A/O

Example:

```
FullMax(measurement)# show afe-rssi
afe-rssi 80 dBm
FullMax(measurement)# track -r4 -i30 afe-rssi
afe-rssi 80 dBm
afe-rssi 80 dBm
afe-rssi 80 dBm
afe-rssi 80 dBm
```

**5.3.5 ss-private**

The bs-private group define the commands that monitors or updates the private mib related information. The user upon entering the ss-private command locks into the CLI to execute commands only related to BS private mib.

```
FullMax#
```

```
FullMax# ss-private
FullMax(ss-private)#
```

Command	Purpose	Privileges
FullMax# ss-private	The command allows the user to lock into the group ss-private and execute all the commands related to the group.	A/O

### ss-rx-amc-count-table

This table contains statistical information that can be used to characterize the adaptive modulation and coding performance in the uplink.

Command	Purpose	Privileges
FullMax(ss-private)# show ss-rx-amc-count-table <diuc-index>	This command shows all the parameters and their values for RxAmcCountTable..	A/O

Example:

```
FullMax(ss-private)# show ss-rx-amc-count-table 1
  ss-rx-octets for diuc-index 1 is 0
  ss-rx-packets for diuc-index 1 is 0
  ss-tx-erroredpackets for diuc-index 1 is 0
```

### ss-rx-diuc-index

The Downlink Interval Usage Code indicates the uplink burst profile in the UCD message.

Command	Purpose	Privileges
FullMax(ss-private)# show ss-rx-diuc-index	This command allows the users to display the available diuc-index	A/O

Example:

```
FullMax(ss-private)# show ss-rx-diuc-index
  ss-ul-diuc-index: 1
  ss-ul-diuc-index: 2
  ss-ul-diuc-index: 3
  ss-ul-diuc-index: 4
  ss-ul-diuc-index: 5
  ss-ul-diuc-index: 6
  ss-ul-diuc-index: 7
  ss-ul-diuc-index: 8
  ss-ul-diuc-index: 9
  ss-ul-diuc-index: 10
```

### ss-rx-octets

This object counts the number of octets received in the downlink using the downlink burst profile indexed by diuc-index.

Command	Purpose	Privileges
FullMax(ss-private)#show ss-rx-octets <diuc-index>	This command allows the user to display the number of octets received in DL burst profile indexed by diuc-index.	A/O

Example:

```
FullMax(ss-private)# show ss-rx-octets 1
ss-rx-octets for diuc-index 1 is 0
```

### ss-rx-packets

This object counts the number of packets received in the downlink using the downlink burst profile indexed by

Command	Purpose	Privileges
FullMax(ss-private)#show ss-rx-packets <diuc-index>	This Command allows the user to display the number of packets received in DL burst profile indexed by diuc-index	A/O

Example:

```
FullMax(ss-private)#show ss-rx-packets 1
ss-rx-packets for diuc-index 1 is 0
```

### ss-rx-erroredPackets

Command	Purpose	Privileges
FullMax(ss-private)#show ss-rx-erroredPackets <diuc-index>	This command allows the use rto display the number of errored packets indexed by diuc-index	A/O

Example:

```
FullMax(ss-private)# show ss-rx-erroredPackets 1
ss-tx-errored packets for uiuc index 1 is 0
```

### ss-tx-amc-count-table

This table contains statistical information that can be used to characterize the adaptive modulation and coding performance in the downlink.

Command	Purpose	Privileges
FullMax(ss-private)#show ss-tx-amc-count-table <uiuc-index>	This command allows the user to display the parametes and values of the TxAmcCount Table.	A/O

Example:

```
FullMax(ss-private)# show ss-tx-amc-count-table 1
ss-tx-octets for uiuc-index 1 is 0
```

ss-tx-packets for uiuc-index 1 is 0

**ss-tx-uiuc-index**

The Uplink Interval Usage Code indicates the uplink burst profile in the UCD message.

Command	Purpose	Privileges
FullMax(ss-private)# show ss-tx-uiuc-index	This command allows the user to display available uiuc- indices	A/O

Example:

```
FullMax(ss-private)# show ss-tx-uiuc-index
ss-dl-uiuc-index: 0
ss-dl-uiuc-index: 1
ss-dl-uiuc-index: 2
ss-dl-uiuc-index: 3
ss-dl-uiuc-index: 4
ss-dl-uiuc-index: 5
ss-dl-uiuc-index: 6
ss-dl-uiuc-index: 7
ss-dl-uiuc-index: 8
ss-dl-uiuc-index: 9
ss-dl-uiuc-index: 10
ss-dl-uiuc-index: 11
ss-dl-uiuc-index: 12
```

**ss-tx-octets**

This object counts the number of octets transmitted in the uplink using the uplink burst profile indexed by uiuc-index

Command	Purpose	Privileges
FullMax(ss-private)#show ss-tx-octets <uiuc-index>	This command allows the user to display the Octets indexed by uiuc-index.	A/O

Example:

```
FullMax(ss-private)# show ss-tx-octets 1
ss-tx-octets for uiuc-index 1 is 0
```

**ss-tx-packets**

This object counts the number of packets transmitted in the uplink using the uplink burst profile indexed by uiuc-index .

Command	Purpose	Privileges
FullMax(ss-private)#show ss-tx-packets <diuc-index>	This command allows the user to display the packets in UL burst indexed by uiuc-index	A/O

Example:

```
FullMax(ss-private)# show ss-tx-packets 1
    ss-tx-packets for uiuc-index 1 is 0
```

### cmn-sf-table

This Table measures service flow traffic

Command	Purpose	Privileges
FullMax(ss-private)# show cmn-sf-table <sfid>	This command allows the user to display the parameters and values for cmn-sf-table for this SFID	A/O

Example:

```
FullMax(ss-private)#show cmn-sf-table 2001
    sf-total-octets for sfid 2001 is 0
    sf-total-pkts for sfid 2001 is 0
    sf-errored-pkts for sfid 2001 is 0
    sf-frag-orig for sfid 2001 is 0
    sf-frag-total for sfid 2001 is 0
    sf-missing-frag for sfid 2001 is 0
```

### cmn-pm-sfid

Command	Purpose	Privileges
FullMax(ss-private)#show cmn-pm-sfid	This command allows the user to display the available SFID.	A/O

Example:

```
FullMax(ss-private)# show cmn-pm-sfid
    cmn-pm-sfid 2001
    cmn-pm-sfid 2002
    cmn-pm-sfid 3000
```

### sf-total-octets

This determines the total octets received/transmitted on this service flow.

Command	Purpose	Privileges
FullMax(ss-private)#show sf-total-octets <sfid>	This command allows the user to display the total octets received or transmitted for this service-flow.	A/O

Example:

```
FullMax(ss-private)# show sf-total-octets 2001
    sf-total-octets for sfid (2001) : 1000
```

**sf-total-pkts**

This determines the total number of packets received /transmitted on this service flow.

Command	Purpose	Privileges
FullMax(ss-private)#show sf-total-pkts <sfid>	This Command allows the user to display the total packets received or trasmitted for this service-flow	A/O

Example:

```
FullMax(bs-private)#show sf-total-pkts 2001
sf-total-pkts for sfid (2001): 22
```

**sf-errored-pkts**

This determines number of packets that were dropped due to missing fragments / bad ARQ blocks.

Command	Purpose	Privileges
FullMax(ss-private)#show sf-errored-pkts <sfid>	This Command Allows the user to display the total errored packets received or transmitted for this service flow.	A/O

Example:

```
FullMax(ss-private)# show sf-errored-pkts 2001
sf-errored-pkts for sfid (2001): 1
```

**sf-frag-orig**

This determines number of fragments / ARQ blocks originally transmitted/received on this service flow.

Command	Purpose	Privileges
FullMax(ss-private)# show sf-frag-orig <sfid>	This Command allows the user to display the total fragments received or transmitted originally on this service flow.	A/O

Example:

```
FullMax(ss-private)# show sf-frag-orig 2001
sf-frag-orig for sfid (2001) : 2
```

**sf-frag-total**

This determines total fragments / ARQ blocks transmitted / received on this service flow.

Command	Purpose	Privileges
FullMax(ss-private)# show sf-frag-total <sfid>	This Command allows the user to display the total fragments received or transmitted on this service flow.	A/O

Example:

```
FullMax(ss-private)# show sf-frag-total 2001
sf-frag-total for sfid (2001): 2
```

### sf-missing-frag

#### Description

This determines the number of missing received fragments received or NACK.

Command	Purpose	Privileges
FullMax(ss-private)# show sf-missing-frag <sfid>	This command allows the user to display the number of missing received fragments or NACK.	A/O

Example:

```
FullMax(ss-private)# show sf-missing-frag 2001
sf-missing-frag for sfid (2001): 1
```

### 5.3.6 ss-chconfig

The ss-chconfig group define the commands that monitors or updates the channel config MIB related information. The user upon entering(locking) the ss-chconfig group can execute commands only related to ss-chconfig group.

```
FullMax#
FullMax# ss-chconfig
FullMax(ss-chconfig)#
```

Command	Purpose	Privileges
FullMax#ss-chconfig	The command allows the user to lock into the group ss-chconfig and execute all the commands related to the group.	A/O

### Channel-config-table

Each entry in the table contains optional channels configuration. A table can include a single active channel configuration and some inactive channel configurations A MS when searching for a new BS will scan the table for available channels.



Command	Purpose	Privileges
FullMax(ss-chconfig)#show ss-channel-config-table <chn-index>	The command allows the user display the channel configuration based on index. A/O	A/O

Example:

```
FullMax(ss-chconfig)# show channel-config-table 1
    center-channel-frequency for chn-index 1 is 12000
Hz
    center-bandwidth for chn-index 1 is 120 Hz
    center-config-status for chn-index 1 is ACTIVE
```

### Channel-config-index

Index in channel-config-table.

Command	Purpose	Privileges
FullMax(ss-chconfig)# show channel-config-index	The command allows the user display all the channel configuration indices	A/O

Example:

```
FullMax(ss-chconfig)# show channel-config-index
    Channel-config-index 1
    Channel-config-index 2
```

### Center-channel-frequency

sets an optional center frequency for the transmitter and receiver Units are in Hz

Command	Purpose	Privileges
FullMax(ss-chconfig)# show center-channel-frequency <chn-index>	The command allows the user display center frequencies for the given index.	A/O

Example:

```
FullMax(ss-chconfig)# show center-channel-frequency 1
    center-channel-frequency for chn-index 1 16000 Hz
```

### Channel-bandwidth

sets the bandwidth of the channel in the associated center frequency Units are in Hz.

Command	Purpose	Privileges
FullMax(ss-chconfig)#show center-bandwidth <chn-index>	The command allows the user display bandwidth for the given index.	A/O

Example:

```
FullMax(ss-chconfig)# show center-bandwidth 2
```

center-bandwidth for chn-index 2 is 1600 Hz

### sampling-clock

The channel sampling clock

Command	Purpose	Privileges
FullMax(ss-chconfig)#show sampling-clock <chn-index>	The command allows the user display sampling clock time for the given index.	A/O

Example:

```
FullMax(ss-chconfig)# show sampling-clock 2
sampling-clock for chn-index 2 is 16
```

### filter-id

The ID of the PHY filter to use for this channel.

Command	Purpose	Privileges
FullMax(ss-chconfig)#show filter-id <chn-index>	The command allows the user display filter id for the given index.	A/O

Example:

```
FullMax(ss-chconfig)# show filter-id 2
filter-id for chn-index 2 is 16
```

### Channel-config-status

Indicates the current state of this entry

inactiveOption(0) - indicates that this entry is a scanning option that is not currently in use.

currentActive(1) - indicates that this entry is the current channel channel configuration

Command	Purpose	Privileges
FullMax(ss-chconfig)#show channel-config-status <chn- index>	The command allows the user display the current status for the given index	A/O

Example:

```
FullMax(ss-chconfig)# show channel-config-status 2
channel-config-status for chn-index 2 is ACTIVE
```

### channel-row-status

This determines the row status of the Channel config table for this config-index.

Command	Purpose	Privileges
FullMax(ss-chconfig)#show	The command allows the user	A/O

channel-row-status	display the row status for the given index	
--------------------	--	--

Example:

```
FullMax(ss-chconfig)# show channel-row-status 2
channel-row-status for chn-index 2 is ACTIVE
```

### 5.3.7 ss-trap

The trap-ctrl-register group defines the commands that monitors or updates the trap mib related information. The user upon entering the trap-ctrl-register command locks into the CLI to execute commands only related to BS and MS trap control register mib.

```
FullMax#
FullMax# ss-trap
FullMax (ss-trap) #
```

Command	Purpose	Privileges
FullMax# ss-trap	The command allows the user to lock into the group ss-trap and execute all the commands related to the group	A/O

Within this group we can get the following parameters.

SstrapControlRegister  
PriTrapControlRegister.

#### ss-trap-control-register

The parameter is used to enable or disable the SS traps.

Command	Purpose	Privileges
FullMax(ss-trap)# set ss-trap-control-register <val> (enable or disable)	The command allows the user to enable or disable the SS traps	A
FullMax(ss-trap)#show ss-trap-control-register	The command allows the user display the enable or disable status of the SS traps.	A/O

Example:

```
FullMax(ss-trap)# set ss-trap-control-register 0
Updated ss-trap-control-register 0 (disable)
FullMax(ss-trap)# show ss-trap-control-register
ss-trap-control-register 0 (disable)
```

#### Rssi-low-threshold

Low Rssi threshold for generating the RSSI alarm trap.

Command	Purpose	Privileges
FullMax(ss-trap)# set rssi-low-threshold <val>	The command allows the user to update the lower threshold for generating RSSI alarm trap	A
FullMax(ss-trap)# show rssi-low-threshold	The command allows the user to display the lower threshold for generating RSSI alarm trap	A/O

Example:

```
FullMax(ss-trap)# set rssi-low-threshold 34
Updated rssi-low-threshold 34 dBm
```

```
FullMax(ss-trap)# show rssi-low-threshold
rssi-low-threshold 34 dBm
```

### Rssi-high-threshold

High Rssi threshold for generating the RSSI alarm trap.

Command	Purpose	Privileges
FullMax(ss-trap)# set rssi-high-threshold <val>	The command allows the user to update the higher threshold for generating RSSI alarm trap	A
FullMax(ss-trap)# show rssi-high-threshold	The command allows the user to display the higher threshold for generating RSSI alarm trap.	A/O

Example:

```
FullMax(ss-trap)# set rssi-high-threshold 34
Updated rssi-high-threshold 54 dBm
```

```
FullMax(ss-trap)# show rssi-high-threshold
rssi-high-threshold 54 dBm
```

### private-trap-ctrl

The object is used to enable or disable Base Station traps From left to right, the set bit indicates the corresponding Base Station trap is enabled.

```
{ wmanPriPowerStatusChange (0), wmanPriFanStatusChange (1),
wmanPriTemperatureChange (2), wmanPriConfigChange (3),
wmanPriAfeBitError (4) }
```

Command	Purpose	Privileges
FullMax(ss-trap)#set private-trap-ctrl <val>	This command allows the user to display the value of the private trap control	A

<pre>FullMax(ss-trap)#show private-trap-ctrl</pre>	<p>This command allows the user to display the value of the private trap control</p>	<p>A/O</p>
--	--	------------

Example:

```
FullMax(ss-trap)# set private-trap-ctrl 21
  updated private-trap-ctrl
    { wmanPriPowerStatusChange(0),
      wmanPriPowerStatusChange(2),
      wmanPriPowerStatusChange(4) }
```

```
FullMax(ss-trap)# show private-trap-ctrl
  private-trap-ctrl enabled
    { wmanPriPowerStatusChange(0),
      wmanPriPowerStatusChange(2),
      wmanPriPowerStatusChange (4) }
```