



ADMINISTRATOR GUIDE

July 2020 | 3725-85483-003A

Poly OBi Devices

OBi3 and OBi5 Series

GETTING HELP

For more information about installing, configuring, and administering Poly/Polycom products or services, go to Polycom Support.

Plantronics, Inc. (Poly – formerly Plantronics and Polycom)
345 Encinal Street
Santa Cruz, California
95060

© 2020 Plantronics, Inc. All rights reserved. Poly, the propeller design, and the Poly logo are trademarks of Plantronics, Inc. All other trademarks are the property of their respective owners.

Contents

Before You Begin	6
Audience, Purpose, and Required Skills	6
Get Help	6
Related Poly and Partner Resources	6
The Polycom Community	7
Documentation Feedback	7
Notational Conventions	7
Canonical Fashion	7
Literal Fashion	8
Boolean Values	8
Multiple Choice Values	8
Parameter Values	8
Getting Started with Poly OBi Devices	9
Product Overview	9
OBi3 and OBi5 Device Features	9
Key Features	9
Robust Telephony Features	10
Powerful Call Routing and Voice Service Features	10
LED Description and Behavior	10
Power Connection	11
USB Connections	11
Ethernet Connections, Setup, and Configuration	11
Phone Ports Setup and Configuration	11
LINE Port Setup and Configuration (Only in OBi312)	12
Primary Line Configuration	12
Configuration and Management Interfaces	13
Configure Your Device Using the IVR System	13
Configure Basic Device Settings	13
Configure System Settings	14
System-Level Configuration Options	15
Configure Network Settings	16
Router Related Configuration Options (OBi302/312/504/508)	18
Configure SIP Service Provider Settings	19

Configure OBiTalk Settings	22
Auto Attendant Configuration Options	24
LINE (FXO) Port Configuration Options for OBi312	24
Customized AA Prompt Recording Options	25
Remote Provisioning	25
Zero-Touch, Massive Scale Remote Provisioning	27
Web Server-Based Local Configuration	27
Local Device Update and Management	28
Update the Firmware	28
Customized AA Prompts Backup and Restore	29
Configuration Backup and Restore	29
Reset Configuration to Factory Default	30
Device Interface	31
Phone Port Setup and Configuration	31
Using the Device as a Paging System	32
Primary Line	32
IP Routing and LAN Switching Features	33
IP Routing Features	33
LAN Switching Features	36
Phone-to-Phone Collaborative Features (OBi504/508)	36
Star Codes	37
Controlling Calls Using Star Codes	37
Star Code List	37
Star Code Script Variables	39
Star Code Script Actions (ACT)	40
Star Code Script Format	41
Star Code Script Examples	41
Status Pages	43
System Status Page	43
WAN Status Section	43
WiFi Status Section	43
Product Information Section	43
OBiBT Bluetooth Service Status Section	44
SP _n Service Status Section	45
OBiTALK Service Status Section	45
LAN Status Page	45
Call Status Page	45

Call History Page	46
Services, Phone and Line Status (OBi312)	46
Device Settings	47
OBiBlueTooth Settings	47
OBiBlueTooth Call Features	47
Phone Interface Features	47
Repeat Dialing Service	47
Audio Line In	48
Audio Line Out	48
Codec Profile Features	49
Tone and Ring Patterns	49
Tone Profile Features	49
Tone Examples	51
Ring Patterns	52
User Settings	53
Speed Dial Numbers	53
User-Defined Digit Maps	54
Network Settings	55
WAN (Internet) Settings	55
LAN Settings	55
DHCP Reservation	55
Firewall and DMZ	55
Port Forwarding	55
QoS Settings	56
Polycom Wi-Fi Wireless Network Adapter	56
WiFi Scan	56
Network Settings	57
Call Routing	58
Inbound Call Route Configuration	58
InboundCallRoute Examples	60
Outbound Call Route Configuration	60
OutboundCallRoute Examples	62
Digit Map Configuration	62
Digit Map Rules and Elements	62
Matching Against Multiple Rules in a Digit Map	64
User-Defined Digit Maps	67
OBi Call Routing and Digit Map	68

Trunks, Endpoints, and Terminals	68
Supported Two-Way Call Bridges	68
Hunt Groups	70
Hunt Group Examples	71
Routing Incoming Calls to Hunt Groups	72
SIP Trunking	72
SIP Trunk Configuration	72
Call Forward Settings Per Phone Port	74
DND Setting Per Phone Port	74
Using Star Codes to Change Per-Phone-Port Call Forward and DND Settings	75
MWI Notification Routing	75
Virtual Stacking	76
Case 1: All Units Having Static or Reserved IP Addresses On the Same Subnet ...	76
Case 2: MU/EU Located Anywhere On the Internet	80
Service Providers	85
SIP Service Provider Features	85
SIP Registration	85
SIP Outbound Proxy Server	86
DNS Lookup of SIP Servers	86
NAT Traversal Considerations	87
SIP Proxy Server Redundancy and Dual REGISTRATION	87
SIP Privacy	88
STUN and ICE	89
ITSP Driven Distinctive Ringing	89
RTP Statistics – the X-RTP-Stat Header	90
Media Loopback Service	90
Using SP n as a Proxy for a SIP IP Phone	91
OBiTALK Service Settings	92
Auto Attendant Service	92
AA Callback Service	92
User-Recorded Prompts	93
Customizing AA Prompt Lists	93
Voice Gateways	94
Trunk Groups	95
Status Parameters	96
System Status Settings	96
Call Status Settings	98
Call History	99
SP Services Stats Settings	99

System Management Parameters	100
WAN Settings	100
Auto Provisioning Settings	104
\$MACRO Expansion Supported by the Device	120
Device Admin Settings	121
Device Update	124
Firmware Update	124
Backup Configuration	125
Restore Configuration	125
Reset Configuration	126
Service Providers Parameters	126
ITSP Profile X (X = A, B, C, D, E, F, G, H)	126
General Settings	126
SIP Settings	128
RTP Settings	135
Voice Services	137
SPn Service Settings (n = 1, 2, 3, 4, 5, 6, 7, 8)	137
OBiTALK Service Settings	143
Gateway Settings	146
Page Group Settings	147
DECT Wireless	148
System Settings	148
Registration Settings	149
Handsetn (n = 1, 2, ..., 9, 10) Settings	149
Codec Profiles	155
Codec Profile X Web Page (X = A, B)	155
Tone Settings	158
Tone Profile X Web Page (X = A, B)	158
Ring Settings	161
Ring Profile X Web Page (X = A, B)	161
Star Codes	162
Star Code Profile X Web Page (X = A, B)	163
User Settings	164
Speed Dial Settings	164
User-Defined Digit Maps Settings	165

Before You Begin

This guide describes how to administer, configure, and provision Polycom OBi Voice Service Bridge and Analog Terminal Adapter (ATA) Devices in the OBi3 and OBi5 Series.

Audience, Purpose, and Required Skills

This guide is for a technical audience composed of cloud service providers and managed service VARs. You must be familiar with the following concepts before beginning:

- Current telecommunications practices, protocols, and principles
- Telecommunication basics, video teleconferencing, and voice or data equipment
- OpenSIP networks and VoIP endpoint environments

Get Help

For more information about installing, configuring, and administering Polycom products, see **Documents & Software** at the [Poly Online Support Center](#).

Related Poly and Partner Resources

See the following sites for information related to this product:

- The [Poly Online Support Center](#) is the entry point to online product, service, and solution support information including **Licensing & Product Registration**, **Self-Service**, **Account Management**, **Product-Related Legal Notices**, and **Documents & Software** downloads.
- The [Polycom Document Library](#) provides support documentation for active products, services, and solutions. The documentation displays in responsive HTML5 format so that you can easily access and view installation, configuration, or administration content from any online device.
- The [Polycom Community](#) provides access to the latest developer and support information. Create an account to access Poly support personnel and participate in developer and support forums. You can find the latest information on hardware, software, and partner solutions topics, share ideas, and solve problems with your colleagues.
- The [Polycom Partner Network](#) are industry leaders who natively integrate the Poly standards-based RealPresence Platform with their customers' current UC infrastructures, making it easy for you to communicate face-to-face with the applications and devices you use every day.
- The [Polycom Collaboration Services](#) help your business succeed and get the most out of your investment through the benefits of collaboration.

Notational Conventions

This guide provides device configuration parameters and their values in either

- Canonical fashion
- Literal fashion

Both notational conventions point to the same parameters, but their appearances are different.

The canonical fashion simplifies locating parameters on the device's native web portal or on OBiTALK.com.

The literal fashion is required when provisioning or writing OBIPhoneXML apps.

Canonical Fashion

This example shows the format of the canonical fashion.

- **Parameter Group Name::ParameterName** = Parameter Value {replace with actual value}

The **Parameter Group Name** is the heading of the parameter group on the left side panel of the device local configuration or OBiTALK Configuration web page. This string may contain spaces. When a group heading has more than one level, each level is separated with a –, such as:

- **Services Providers - ITSP Profile A – SIP:**

The **ParameterName** is the name of the parameter as shown on the web page and **MUST NOT CONTAIN ANY SPACES**. **Parameter Group Name** and **ParameterName** are separated by two colons (::), as shown in the first example above.

The **Parameter Value** is the literal value to assign to the named parameter and may contain spaces. You can omit **Parameter Group Name** or its top-level headings when the context is clear. For example:

- **SP1 Service::AuthUserName** = 4082224312
- **ITSP Profile A - SIP::ProxyServer** = sip.myserviceprovider.com
- **ProxyServerPort** = 5082

Literal Fashion

These examples show the format of the literal fashion. The literal fashion is used when provisioning or writing OBIPhoneXML apps.

- **ParameterGroupName.ParameterName**.Parameter Value {replace-with-actual-value}
- **Parameter.Group.Name.ParameterGroupName.ParameterName**.Parameter Value

The **ParameterGroupName** is the name of the first parameter group in literal fashion. This string **MUST NOT CONTAIN ANY SPACES**, and always is terminated with a period, as shown. More than one **ParameterGroupName** may be used. The **ParameterGroupName** is case-sensitive.

The **ParameterName** is the name of the parameter, and always is terminated with a period, as shown. This string **MUST NOT CONTAIN ANY SPACES**. The **ParameterName** is case-sensitive.

The **Parameter Value** is the literal value to assign to the named parameter and may contain spaces. The **Parameter Value** isn't case-sensitive, but it **MUST EXACTLY MATCH** the value when one or more choices are available.

When using the literal fashion in your XML, you need to exactly match the text string for ***ParameterGroupName.ParameterName***.Parameter Value, but text formatting such as bold face isn't required and will be removed when your script or app is processed.

Boolean Values

You can identify parameters that take a Boolean value on your device's configuration web pages by a check box next to the parameter name. Throughout this guide, we may loosely refer to a Boolean value as "enable/disable" or "yes/no", but the only valid Boolean parameter values to use in a device configuration file is either `true/false` or `True/False` (case-sensitive). This is equivalent to selecting or clearing the check box on the configuration web pages.

Multiple Choice Values

You must provision parameters that take one of several valid options from a drop-down list on the device message with string values that match exactly one of those choices. Otherwise, the device uses the default choice. Matching the provisioned value against valid strings is case-sensitive and doesn't allow extra spaces. When a choice must be selected, the device web page provides a drop-down menu for that parameter. Copy that value into your provisioning script.

Parameter Values

When entering a parameter value from the web page or via provisioning, avoid adding extra white spaces before or after the parameter value. If the value is a comma-separated list of strings or contains attributes after a comma or semicolon, avoid adding extra white space before and after the delimiter.

For example: **CertainParameter** = 1,2,3,4;a;b;c

If a parameter value can include white spaces, such as **X_DisplayLabel**, use just a single space and no extra space before and after the value.

For example: **X_DisplayLabel** = My New Service

Getting Started with Poly OBi Devices

The terms *the device* and *your device* refer to any of the Poly OBi ATA devices in the OBi 3 and OBi5 Series. Unless specifically noted in this guide, all device models operate in similar ways.

Product Overview

This guide describes the following OBi devices:

- OBi300 Series: OBi300/302/312
- OBi504
- OBi508

OBi3 and OBi5 Device Features

Built with a high-performance system-on-a-chip platform to ensure high-quality voice conversations, OBi devices are dedicated systems targeted at applications for Voice over IP services. OBi devices have high availability and reliability because they're always on to make or receive calls.

Key Features

OBi3 and OBi5 Series Devices

OBi Device Model	VoIP Accounts	PDMS-SP Support	PHONE Ports	LINE Ports	GigE Ports	Fast E-net Ports	USB Ports	Audio Line In	Audio Line Out	PSTN Failover Ports
OBi300	4	Yes	1	0		1	1			
OBi302	4	Yes	2	0		2	1			
OBi312	4	Yes	1	1		2	1			
OBi504	9	Yes	4	0	1*	3*	3	1	1	1**
OBi508	9	Yes	8	0	1*	3*	3	1	1	1**

* OBi3 and OBi5 Series devices can be configured as a router or as a switch. In router mode, the GigE port functions as the WAN port, and the three Fast Ethernet ports as three LAN switch ports. In switch mode, all four ports function as switch ports.

** For failover during power outage only.

Robust Telephony Features

The OBi 3 and OBi5 Series devices have the following robust telephony features:

- Message waiting indication – visual and tone based
- Speed dialing of 99 OBi endpoints or numbers
- Three-way conference calling with local mixing
- Hook flash event signaling
- Caller ID – name and number
- Call waiting
- Call forward unconditional
- Call forward on busy
- Call forward on no answer
- Call transfer
- Anonymous call
- Block anonymous call
- Do not disturb
- Call return
- Repeat dialing
- Collaboration – PBX-like call features among all phone ports
- Paging through the Line Out port (OBi504/508)
- Music On Hold (OBi504/508)
- Multilevel Auto Attendants with fully customizable prompts (OBi504/508)

Powerful Call Routing and Voice Service Features

The OBi3 and OBi5 Series devices have the following powerful call routing and voice service features:

- SIP support for voice and fax over IP from internet telephony service providers
- PDMS-SP managed VoIP network for OBi endpoint devices and applications
- High-quality voice encoding using G.711, G.726, G.729, and iLBC algorithms
- Recursive digit maps and associated call routing (outbound, inbound)
- Service continuity in case of power or network failure (configurable)

LED Description and Behavior

Depending on the device model, there are 3 -14 LEDs on the front of the device. They provide you with a visual indication of the working order and general status of key functional aspects of the device. Under normal operating conditions, the LEDs show green (solid or blinking) signals.

Power Connection

Connect the supplied 12-volt power adapter to the device and the wall outlet or working power strip. Only use the power adapter supplied with the original packaging to power the device. Use of any power adapter other than what was provided with the device will void the warranty and may cause the unit to not function at all or cause undesired operation.

USB Connections

The OBi3 Series has one USB 2.0 ports whereas OBi5 has three USB ports.

At present the OBi 3 and OBi5 Series VoIP adapters only recognize Polycom OBiBT Bluetooth adapters. You can connect one OBiBT Bluetooth adapter to each of USB Port 1 and USB Port 2. The corresponding services are referred to as **OBiBlueTooth 1** and **OBiBlueTooth 2** in the device configuration.

The OBi3 Series supports USB WiFi dongles with OBiWiFi dongles only. OBi5 Series does not support WiFi.

Ethernet Connections, Setup, and Configuration

Connect one end of an Ethernet cable to the device's Ethernet Port 1 (which is the GigE port), and the other end of the cable to an Ethernet port on your Internet router or Ethernet switch. By default, the device requests an IP address, a DNS, and Internet (WAN) Gateway IP addressing via DHCP.

The OBi3 and OBi5 devices can be configured either as a:

- WAN Port (to Uplink) and LAN Port(s) as a router or a switch using the parameter **DeviceInfo.LAN.OperationMode**
- Switch with 4 LAN switching ports via the parameter **DeviceInfo.LAN.UseWANPortAsLANPort**

To connect to the network when the device is configured as a switch, connect one end of an Ethernet cable to any one of the four OBi3 and OBi5 Ethernet ports and the other end of the Ethernet cable to an Ethernet port on your Internet router or Ethernet switch. To connect to the network when the device is configured as a router, connect one end of an Ethernet cable to Ethernet Port 1 of the OBi3 and OBi5 and the other end of the Ethernet cable to an Ethernet port on your Internet router or Ethernet switch.

Phone Ports Setup and Configuration

A phone has a very basic UI (user interface) for I/O (Input / Output) of signaling or control messages. Similar to a land line phone, the OBi device's Phone port supports input and output signaling and control messages.

Connect your touch-tone phones to your device's Phone ports. The OBi3 and OBi5 devices support only touch-tone telephones. Pulse dialing isn't supported.

An OBi Phone port can select from the following services to complete a call:

- SP1–4 for OBi3 Series
- SP1-9 for OBi5 Series
- OBiTALK (PP1)
- OBiBlueTooth 1 (BT1)
- OBiBlueTooth 2 (BT2 only for OBi5)

See the [Phone Port Setup and Configuration](#) section for more information.

LINE Port Setup and Configuration (Only in OBi312)

Connect an active analog telephone (POTS) line to your device's LINE port, if so equipped.

The device LINE port supports the following outputs:

- DTMF/Tone
- On Hook
- Off Hook
- Hook Flash

The device LINE port assumes a call is Connected on the following conditions:

- End of dialing for outbound calls
- Off-hook for inbound calls

The device LINE port assumes a call is Disconnected on the following conditions:

- Power Down
- CPC
- Long Silence
- Disconnect Tone



The device LINE port and system logic won't attempt to invoke or interpret PSTN supplementary services. This is directly between the user and the Phone Company. To signal hook-flash to the PSTN Line during a call from the phone attached to the PHONE port, the **Phone** **PORT::HookFlashHandling** parameter must be set to `Send Flash Hook to PSTN`.

Primary Line Configuration

Your Primary Line is the service for making an outbound call that doesn't require an access code prefix (such as **1 or **9) when dialing. It's the default service used to make the call when no explicit access code prefix is entered.

You can select a service or Trunk Group (TG) as the Primary Line. The following list summarizes the choices available for selection as the primary line:

- SP Service
- OBiTALK Service
- OBiBlueTooth
- Trunk Group 1 or 2 (OBi5 Series Only)

An SP service can be a SIP-based service. In addition, all device models come with the OBiTALK (peer-to-peer) service.

For more information on Primary Line features, see the [Primary Line](#) section.

Configuration and Management Interfaces

The OBi device provides these interfaces for local configuration and management:

- Device local interface
- OBITALK.com system web interface

Additionally, remote configuration and management is possible using the PDMS-SP web page.

Configure Your Device Using the IVR System

The OBi device uses the IVR system for both its configuration and normal functionality. Access the IVR system to receive verbal prompts and information from the device (such as the device IP address).

Note the following information regarding the IVR system:

- If a setting change requires a reboot, the system reboots automatically when you quit the IVR system.
- You can access the next menu of the IVR system or invoke a command without waiting for the previous announcement to end.

Configure Basic Device Settings

Use the IVR system's main menu to configure your device's basic settings or to access additional configuration menus.

To configure basic device settings:

- 1 Dial *** from the handset.
- 2 Enter the number for the configuration menu you want to access.

Configure Device Settings

Menu Selection	Setting	Description
1	Basic Network Status	Device IP address and DHCP status.
2	Advanced Network Status	Information on the primary and back-up DNS server and primary and back-up NTP server.
3	Set DHCP	Current DHCP value. • Press 0 to repeat the information. • Press 1 to enter a new value. • Press 2 to set the default value.

Configure Device Settings

Menu Selection	Setting	Description
4	Set IP Address	Current IP address. Note: If you enter a new value (static IP address), DHCP is disabled. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value.
5	Set Password	Current IVR password. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value.
6	Software Update The device plays one of the following messages: Software update available. Press 1 to update software. Software update not available.	If an update is available, press 1 to update the software. The software update process starts as soon as you hang up the phone. Warning: Once the software upgrade process starts, the device's power LED blinks rapidly. Make sure the power and network cable stay connected to the unit until the process is complete.
8	Restore Factory Default	Restores the device to factory default settings. - Press 1 to confirm the factory restore. - Press # to return to the main configuration menu. - Press ## to exit the IVR system.
9	Reboot	Reboots the device. - Press 1 to confirm device reboot. - Press # to return to the main configuration menu. - Press ## or hang up to exit the IVR system.
0	Additional Options	Access other configuration settings for your handset.

Configure System Settings

You can configure system options through the system settings submenu. However, the device doesn't announce the available settings in the submenu.

To configure system settings:

- 1 Dial *****0** from the handset.
- 2 Enter the number for the configuration menu you want to access, followed by the **#** key.

Menu Selection	Setting	Description
1	Firmware Version	Current firmware version. - Press 0 to repeat the information. - Press # to enter another configuration menu selection.
2	IVR Password	Current IVR password. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
3	Debug Level	Current debug level. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
4	Syslog Server IP Address	Current syslog server IP address. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
5	Syslog Server Port	Current syslog server port value. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value of 514. - Press # to enter another configuration menu selection.

System-Level Configuration Options

Additional configuration options are available with the device IVR after pressing *****0**.

There are many additional options beyond the top-level IVR operations options 1 – 8. Unlike the top-level options, however, the list of available additional options in Menu 0 isn't announced.

To access the additional IVR options, do the following:

- 1 Enter *****0#**
- 2 Enter your desired option number, followed by **#**

3 Follow any additional prompts, as desired.

You must enter the corresponding option number (followed by the **#** key) to select the particular option. The following tables list the available additional options (grouped by function):

System-Level Configuration Options

Menu Selection	Settings	Descriptions
1	Firmware Version	Current firmware version. - Press 0 to repeat the information. - Press # to enter another configuration menu selection.
2	IVR Password	Current IVR password. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
3	Debug Level	Current debug level. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
4	Syslog Server IP Address	Current syslog server IP address. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
5	Syslog Server Port	Current syslog server port value. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value of 514. - Press # to enter another configuration menu selection.

Configure Network Settings

You can configure network options through the network settings submenu. However, the device doesn't announce the available settings in the submenu.

To configure network settings:

- 1 Dial *****0** from the handset.
- 2 Enter the number for the configuration menu you want to access, followed by the **#** key.

Configure Network Settings

Menu Selection	Setting	Description
20	DHCP Configuration	Current DHCP configuration value. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
21	IP Address	Current IP address. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
22	Default Gateway	Current default internet gateway. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
23	Subnet Mask	Current subnet mask. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
24	DNS Server (Primary)	Current primary DNS server. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.

Configure Network Settings

Menu Selection	Setting	Description
25	LLDP Discovery (Enable/Disable)	Current LLDP Discovery configuration value. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. Press # to enter another configuration menu selection.
26	NTP Server (Primary)	Current primary NTP server. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.

Router Related Configuration Options (OBi302/312/504/508)**To get additional router-related configuration options:**

- » Dial *****0** from the handset.

Configuration options are available with the device IVR.

Router Related Configuration Options (OBi302/312/504/508 Only)

Menu Selection	Setting	Description
30	Enable Web Management Access from WAN port. The current value is read back. (Effective in Router mode only) 0: Disable access from WAN 1: Enable access from WAN	- Press 1 to enter a new value. - Press 2 to set the default value.
31	Choose Router or Bridge Mode. The current value is read back. 0: Router 1: Bridge	- Press 1 to enter a new value. - Press 2 to set the default value. - Press 0 to repeat the information. - Press # to enter another configuration selection.
32	Enable DHCP Server on LAN side in router mode. The current value is read back. 0: Disable DHCP server 1: Enable DHCP server	- Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. - Press # to enter another configuration selection.

Menu Selection	Setting	Description
33	Enable Firewall in router mode. The current value is read back. 0: Disable firewall 1: Enable firewall	- Press 1 to enter a new value. - Press 2 to set the default value. - Press 0 to repeat the information. - Press # to enter another configuration selection.
38	Enable QoS feature in router mode. The current value is read back. 0: Disable QoS 1: Enable QoS	- Press 1 to enter a new value. - Press 2 to set the default value. - Press 0 to repeat the information. - Press # to enter another configuration selection.

Configure SIP Service Provider Settings

You can configure SIP service provider options through the SIP service provider settings submenu. However, the device doesn't announce the available settings in the submenu.

To configure SIP service provider settings:

- 1 Dial *****0** from the handset.
- 2 Enter the number for the configuration menu you want to access, followed by the **#** key.

SP1 Configuration Options

Menu Selection	Setting	Description
100	Enable Service Provider One (SP1)	Current SP1 value. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
101	Registration State of SP1	SP1 registration state. - Press 0 to repeat the information. - Press # to enter another configuration menu selection.
102	SP1 User ID	SP1 user ID value. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.

SP1 Configuration Options

Menu Selection	Setting	Description
167	SP1 Block Caller ID Enable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
168	SP1 Block Anonymous Call Enable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
172	SP1 Call Forward ALL – Enable / Disable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
173	SP1 Call Forward ALL Number	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
174	SP1 Call Forward on Busy – Enable / Disable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
175	SP1 Call Forward on Busy Number	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
176	SP1 Call Forward on No Answer – Enable / Disable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
177	SP1 Call Forward on No Answer Number	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.

Additional configuration options are available with the device IVR after pressing * * * 0 for SIP service provider 2 (SP2).

SP2 Configuration Options

Selection	Announcement	What Can You Do?
200	Enable Service Provider Two SP2.	Current SP2 value. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
201	Registration State of SP2	SP2 registration state. - Press 0 to repeat the information. - Press # to enter another configuration menu selection.
202	SP2 User ID	SP2 user ID value. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
267	SP2 Block Caller ID Enable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
268	SP2 Block Anonymous Call Enable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
272	SP2 Call Forward ALL – Enable / Disable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
273	SP2 Call Forward ALL Number	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.

SP2 Configuration Options

Selection	Announcement	What Can You Do?
274	SP2 Call Forward on Busy – Enable / Disable	• Press 0 to repeat the information. • Press 1 to enter a new value. • Press 2 to set the default value. • Press # to enter another configuration menu selection.
275	SP2 Call Forward on Busy Number	• Press 0 to repeat the information. • Press 1 to enter a new value. • Press 2 to set the default value. • Press # to enter another configuration menu selection.
276	SP2 Call Forward on No Answer – Enable / Disable	• Press 0 to repeat the information. • Press 1 to enter a new value. • Press 2 to set the default value. • Press # to enter another configuration menu selection.
277	SP2 Call Forward on No Answer Number	• Press 0 to repeat the information. • Press 1 to enter a new value. • Press 2 to set the default value. • Press # to enter another configuration menu selection.

Configure OBiTalk Settings

You can configure OBitalk options through the OBitalk settings submenu. However, the device doesn't announce the available settings in the submenu.

To configure OBitalk settings:

- 1 Dial *****0** from the handset.
- 2 Enter the number for the configuration menu you want to access, followed by the **#** key.

Configure OBitalk Settings:

Menu Selection	Setting	Description
900	Enable OBiTALK Service	Current OBiTALK service value. - Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
901	Registration State of OBiTALK	OBiTALK registration state. - Press 0 to repeat the information. - Press # to enter another configuration menu selection.
967	OBiTALK Block Caller ID Enable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
968	OBiTALK Block Anonymous Call Enable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
972	OBiTALK Call Forward ALL – Enable / Disable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
973	OBiTALK Call Forward ALL Number	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
974	OBiTALK Call Forward on Busy – Enable / Disable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
975	OBiTALK Call Forward on Busy Number	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.

Configure OBitalk Settings:

Menu Selection	Setting	Description
976	OBiTALK Call Forward on No Answer – Enable / Disable	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.
977	OBiTALK Call Forward on No Answer Number	- Press 0 to repeat the information. - Press 1 to enter a new value. - Press 2 to set the default value. - Press # to enter another configuration menu selection.

Auto Attendant Configuration Options

Auto Attendant configuration options are available with the device IVR after pressing *****0**.

Auto Attendant Options

Selection	Announcement	What Can You Do?
80	Enable / Disable Auto Attendant.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.

LINE (FXO) Port Configuration Options for OBi312

LINE (FXO) port configuration options are available for OBi312 with the device IVR after pressing *****0**.

LINE (FXO) Port Configuration Options

Selection	Announcement	What Can You Do?
90	Enable / Disable FXO LINE Port The current value is read back.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
91	FXO State The current state is read back.	Press 0 to repeat the information. Press # to enter another configuration selection.
92	Ring Through Delay The current value is read back.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.

Customized AA Prompt Recording Options

Customized AA prompt recording options are available with the device IVR after pressing * * * 0:

Customized AA Prompt Recording Options

Selection	Announcement	What Can You Do?
1001	Option 1001 current value is: (the recorded prompt)	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection. Note: After pressing 1 to record a new prompt, the device says "Enter value followed by the # key." At that point, you can press any digit (0-9) to start recording, and then press # to end recording. Tips: Leave about 1 second of gap at the end of recording to avoid unintended truncation by the device. After a new prompt is recorded, the device immediately plays back the recorded audio, and then presents the following options: Press 1 to save (saves the recorded prompt permanently in long-term memory). Press 2 to re-enter (discards the last recorded prompt). Press 3 to review. Press # to cancel (discards the last recorded prompt).
	Similarly for Options 1002 through 1020	

With these options you can record as many as 10 prompts, which can be arranged in any combinations and used as customized AA prompts. Each prompt recording is limited to 60 seconds, where the prompt duration is rounded to the nearest number of seconds. A total of 122 seconds is available to store all the recordings. The device reboots automatically when you hang up if any of the prompts have been modified and saved. You can enter a text description for each recorded prompt as a reminder of the contents of the prompt.

Remote Provisioning

Remote Provisioning is the process by which OBi devices download a configuration file from a server, located in the cloud or in the same enterprise.

The configuration file contains all the necessary parameter values for the device to function normally. It also can tell the device to download an additional configuration file from a different URL or to download a different firmware to replace the current one. The configuration file format and parameter naming conventions are proprietary to Poly but are common across all Poly products.

It supports two configuration file formats: A full XML format with the XML tags in full text and a short XML format with the XML tags substituted with a single letter abbreviation. The XML structure and parameter naming convention closely follows TR-069/TR-104.

Similar to the way parameters are grouped under different device configuration web pages, parameters are grouped into a number of configuration objects for remote provisioning.

The corresponding configuration object in a handset configuration XML file is:

VoiceService.1.VoiceProfile.1.Line.1.SIP.

as shown:

```
<Object>
  <Name>VoiceService.1.VoiceProfile.1.Line.1.SIP.</Name>
  <ParameterValueStruct>
    <Name>AuthUserName</Name>
    <Value>john.j.smith@gmail.com</Value>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>AuthPassword</Name>
    <Value>zYz123#$12</Value>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>URI</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_MyExtension</Name>
    <Value>16188</Value>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_XsiUserName</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_XsiPassword</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_XmppDomain</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_XmppUserName</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_ContactUserID</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_EnforceRequestUserID</Name>
```

```
<Value X_UseDefault="Yes"/>
</ParameterValueStruct>
</Object>
```

Note that you must not omit the dot (.) at the end of the object name is part of the name in the XML file. You must use the correct object name to create a valid configuration file for the handset. You can find the object name corresponding to each configuration web page/section listed in the [Parameter Reference](#) section at the end of this guide.

Data model is a collective list of configuration parameters, syntaxes, and valid values for a specific device. You can find the most up-to-date data model for the device online at the following URL:

<http://www1.obitalk.com/Downloads/dev/datamodel/obihai504.xml>

<http://www1.obitalk.com/Downloads/dev/datamodel/obihai508.xml>

Zero-Touch, Massive Scale Remote Provisioning

Zero-Touch or ZT provisioning is a system level approach to deploying and maintaining thousands or millions of devices with high security and control at the device level down to the individual parameter provisioned on each device.

To enable ZT provisioning, customize the **ITSP Provisioning::ConfigURL** parameter, which tells the handset where to download a configuration file. With this parameter configured, the first time a new handset is powered on and connected to the network, it can automatically contact the designated URL to get the initial configuration file.

For more information on using ZT provisioning, contact your Poly sales representative.



Zero-Touch devices must contact OBiTALK.com one time to get the customized values before they can start normal operation. Make sure that the device can access the internet before first use.

Web Server-Based Local Configuration

The device has an integrated device management web server that you can access from a PC or similar device using a browser. Although all popular browsers are tested for compatibility with the device management web server, there may be inconsistencies that arise from time to time. Please contact obi.spsupport@polycom.com if you have any questions about the device management web server and how it appears in your browser window.

Access the Device Management Web Page

- 1 From a phone attached to the device, dial * * * to access the device Config Attendant.
- 2 Choose "1" to hear the IP address of the device read back to you. Write this down.
- 3 Enter the device IP address in a local PC web browser address field:
- 4 When prompted, enter `admin` for user name and `admin` for password.

When you access the device management web page, you are prompted for a user name and password. There are two levels of access to the device web page – User Level and Admin Level. The default user name / password for User Level access is `user / user`. The default user name / password for Admin Level access

is `admin/admin`. The Admin and/or User passwords may have been changed using the device web page, provisioning by a service provider, or via the OBiTALK web portal (Admin only). Please be sure that you have access to the correct Admin or User password before you attempt to log on to the device management web page.

The device management web page is organized into sections to allow for a manageable and compartmentalized approach to configuring the many hundreds of parameters available on the device. Use the expandable / collapsible menu tree on the left side of the page to easily navigate the various configuration parameter sections of the device.



Every configuration page must be submitted individually after changes made on the page. Otherwise those changes are discarded once you navigate to another page. Most changes require a reboot of the unit (by clicking the **Reboot** button) to take effect. However, you may reboot the unit just once after you have made and submitted all the necessary changes on all the pages.

When the device is operating in router mode, the built-in web server can be accessed from the LAN side or the WAN side. While access from the LAN side is always allowed, for security reasons, the access from the WAN side can be disabled by configuration. WAN side access to the web server is disabled by default. You can enable this option on the device web page (from the LAN side), or by using the device configuration IVR (* * * 0 option 30) from an attached telephone.

Local Device Update and Management

You can use the device configuration webpage to manage or update your device firmware.

Update the Firmware

You can upgrade the firmware for your device from the device configuration web page. The firmware file with which you want to upgrade the device must be stored locally on a computer that you can access with a web browser.

Follow these steps to upgrade:

- 1 Select the **System Management – Device Update** menu on the side panel of the web page.
- 2 Specify the path of the firmware file by clicking the **Browse** button in the **Firmware Update** section of the page. This opens a file browser window where you can select the firmware file.
- 3 Upon selection of the firmware file, press the **Update** button to start the upgrade process.

The process takes about 30 seconds to complete. Note that you **MUST NOT** disconnect the power from the device during this procedure. If the new firmware is upgraded successfully, the device reboots automatically to start running the new firmware. Otherwise, the web page shows an error message explaining why the upgrade failed.

Possible Error Messages on Firmware Update Failure

Error Message	Description	Suggested Solution
Firmware Package Checksum Error	A corrupted Firmware package file has been used for the update.	Check the file and / or re-download the firmware package and try again.
System Is Busy	The device is busy because one of the phone services is in an active call or device provisioning is in progress.	Try to update again later.
Firmware Is Not Modified	The device is already running the same firmware as the one selected for update.	No need to upgrade.

Customized AA Prompts Backup and Restore

As many as 10 individual prompts can be recorded through the device IVR interface. These prompts can be backed up into a single file from the web browser. The default name of the file is `backupaa.dat`. The backup file also includes the annotations entered for each recorded prompt.

To restore an AA prompt file onto a device, do it exactly like a firmware upgrade via the web browser, but provide the device with the prompt file instead of a firmware file. The device can detect from the file header that you are trying to upload a prompt file and process the file accordingly.



Warning: All the existing prompts in the device are removed first when applying the backup file. This process cannot be undone.

Configuration Backup and Restore

The current configuration of the device can be backed up and stored as a file in XML format at a user specified location. The default name of the file is `backupxxxxxxxxxxxx.xml`, where the `xxxxxxxxxxxx` represents the MAC address of unit.

When backing up a device's configuration, you may select the following options before starting the backup.

Configuration Backup and Restore

Option	Description	Default Setting
Incl. Running Status	If checked, the values of all status parameters are included in backup file. Otherwise, status parameters are excluded from the backup.	No
Incl. Default Value	If checked, the default values of parameters are included in the backup file. Otherwise, default values are excluded from the backup.	No
Use OBi Version	If not checked, the backup file uses XML tags that are compliant with TR-104 standard. Otherwise, the backup file is stored in an OBi proprietary format where the XML tags are not compliant with TR-104, but the file size is smaller and the file is more readable.	No

When the file browser window pops up for, you can change the file name and choose the location to save the backup file. Note that different web browsers might handle this differently. If the operation is blocked due to the security setting of the web browser, you should change the security setting temporarily to allow this operation to complete.

When restoring the configuration to a previous backup copy, you will need to specify the backup file you want to restore to by selecting the **Browse** button in the **Restore Configuration** section of the web page. Then, select the **Restore** button to start the process. The device automatically reboots after the restoration is complete.



All passwords and PINs are excluded from the backup file. Hence they will not be available to restore. Call history is excluded from the backup, but can be saved as an XML formatted file separately from the Call History web page

Reset Configuration to Factory Default

The device can be reset to the factory default condition. Call history and various statistical information are removed at the same time. Resetting the device configuration should be used with extreme caution as the operation cannot be undone. To do this, press the **Reset** button in the **Reset Configuration** section. A confirmation window pops up. The device then proceeds to reset the configuration once you confirm that this is indeed what you want to do. The device reboots automatically when factory reset completed.

Device Interface

This section provides information on collaborative features from phone-to-phone, end user features and setting up of phone port.

Phone Port Setup and Configuration

A phone has a very basic UI (user interface) for I/O (Input / Output) of signaling or control messages. Similar to a land line phone, the OBi device's Phone port supports input and output signaling and control messages.

Connect your touch-tone phones to your device's Phone ports. The OBi3 and OBi5 series devices support only touch-tone telephones. Pulse dialing isn't supported.

The device Phone port supports input signaling and control messages comprising:

- On Hook
- Off Hook
- Hook Flash
- DTMF tones

The device Phone port supports output signaling and control messages comprising:

- Caller ID/CWCID
- MWI
- DTMF/Tone
- Ring
- Polarity Reversal
- CPC
- Power Denial

Each device Phone port has a Maximum Sessions capacity of two. This isn't configurable. The device Phone ports reply BUSY to a new incoming call when:

- The Phone port already has two calls in session.
- The Phone port is ringing the phone.
- The phone is in a dialing or fast busy "Invalid" state.
- The device is already in a FAX call.

The device Phone ports support Call Waiting when a second call is an inbound call:

- A Hook-Flash (or depressing the Flash button) invokes switching between two calls.
- When the device Phone port goes On-Hook, this ends the current call and invokes a ring for the holding call. The device Phone port supports 3-way Calling when the second call is an outbound call.

On the first Hook-Flash during an active call, the device can make a second outbound call.

On the second Hook-Flash, the first call and the second outbound call are placed in a conference. To remove the second conferenced party, invoke a third Hook-Flash.

When the device goes On-Hook during a 3-way Call, this becomes a transfer when second (outbound) call is ringing or connected. If the second (outbound) call doesn't succeed, then the device Phone port can go to an On-Hook state and ring as the holding call is still on the line, or Hook-Flash to resume the first call.

The device Phone ports can select from the following services for completing a call:

- SP1 Service (SP1)
- SP2 Service (SP2)
- SP3 Service (SP3)
- SP4 Service (SP4)
- SP5 Service (SP5)
- SP6 Service (SP6)
- SP7 Service (SP7)
- SP8 Service (SP8)
- SP9 Service (SP9)
- OBiTALK Service (PP1)
- OBiBlueTooth 1 Service (BT1)
- OBiBlueTooth 2 Service (BT2)

Using the Device as a Paging System

You can connect one of the device's Phone ports to an external PA system via an RJ11-to-Line-Out connector, and enable the Phone port **UseForPagingOnly** parameter. In this configuration, the Phone port is expected to be "off-hook" all the time. The device automatically answers incoming calls. It won't accept call-waiting.

When the Phone port goes from on-hook to off-hook, in case you need to dial * * * to invoke the IVR, the device plays a dial tone for 5 seconds. After 5 seconds, the device turns silent and is ready to accept an incoming call to page.

You can also connect the Audio Line Out port to an external PA system for paging. The Line Out port is referred to as **ao** (which stands for Audio Out) in call routing rules. With the default configuration, a user can broadcast an announcement to the Line Out port by picking up any of the phone ports and dialing **#0**. This is achieved with the rule {#0:ao} in the **OutboundCallRoute** parameter for each phone port. The OBi plays a short beep before playing the caller's announcement.

Similarly, a remote caller can also be routed to page with a similar routing rule in the Inbound Call Route. For example, adding this rule {>100:ao} to the **SP1 Service::X_InboundCallRoute** parameter allows a caller at SP1 to send audio out to the Line Out port by calling the extension 100.

Primary Line

Select the Primary Line for the Phone *n* Port for *n* = 1, 2, ..., 8, and for the AA, respectively, using the **PHONE *n* Port::PrimaryLine** and **Auto Attendant::PrimaryLine** parameters.

An SP service can be a SIP-based service. In addition, all device models come with the free OBiTALK (peer-to-peer) service. In this guide, these voice services are referred to as a trunk and a trunk group (TG), which is a comma-separated ordered list of trunks. If you select a TG for making an outbound call, the device picks the first available member in that trunk group for the call. You can define as many as four TGs in a device. See the [Trunk Groups](#) section for details.

You can make one of the SP Services, OBiBlueTooth 1/2, or TG1/TG2 as the Primary Line for outbound calls. Primary Lines for the Phone ports and the Auto Attendant are configured via the system web interface described in this guide. The OBiTALK Device Configuration VoIP Service Provider set-up screen also gives you the option to select a service or TG as the Primary Line. This list summarizes the choices available for selection as the primary line:

- SP1 Service
- SP2 Service
- SP3 Service
- SP4 Service
- SP5 Service
- SP6 Service
- SP7 Service
- SP8 Service
- SP9 Service
- OBiTALK Service
- OBiBlueTooth 1
- OBiBlueTooth 2
- Trunk Group 1
- Trunk Group 2

If necessary, you can customize service route access codes for calling from the Phone port by modifying the **PHONE *n* Port::DigitMap** and **PHONE *n* Port::OutboundCallRoute** parameters. You can customize service route access codes for calling via the Auto Attendant by modifying the **Auto Attendant::DigitMap** and **Auto Attendant::OutboundCallRoute** parameters.

Occurrences of (Mpli) and pli are substituted internally with the corresponding abbreviated trunk name of the selected primary line.

IP Routing and LAN Switching Features

The device has two Ethernet ports labeled as the Internet port and the LAN port. The device works as a router by default. All the native voice services and features use the WAN port only when the device is in router mode. The device can also be set to work as a 3-port switch (in Bridge mode) by changing its **OperationMode** parameter from `Router` to `Bridge`. One of the switch ports is for device internal use only.

IP Routing Features

In router mode, the network connected to the device Internet Port is the WAN side of the device, and the network connected to the device LAN Port is the LAN side of the device. The WAN side can be connected to another Ethernet switch or directly to an access device such as a cable or DSL modem for Internet

access. The device routes traffic between the LAN side and the WAN side, thus allowing the devices (such as PCs) attached to the LAN side to share Internet access. The device supports subnet masks as large as 255.255.255.0 to accommodate as many as 253 IP addresses on its LAN side subnet.

In addition to being a NAT (Network Address Translation) router, the device includes a DHCP server, a DNS forwarder, and a basic firewall. It supports port forwarding, DMZ, QoS, and VLAN (802.1Q). The maximum routing throughput between the WAN and the LAN side is approximately 30 Mbps. This speed can be achieved when there are no active calls in the system. Otherwise, throughput is limited to a slower speed to accommodate the load for voice processing. Note that if the WAN side is connected to an Internet access device directly, then the throughput could be further limited by the speed of the Internet uplink and downlink.

The device acquires its WAN side IP address using one of the following methods:

- Static Address Assignment
- DHCP
- PPPoE

By default, the device acquires its WAN side IP address using DHCP. Also by default, the device's own DHCP server is enabled to support LAN side clients such as PCs. The default LAN side IP address of the router is 192.168.10.1.

Incoming packets received from the WAN side are forwarded by the router according to the following flow:

- If the firewall is enabled, discard the packet if it's rejected by any one of the active firewall components.
- If the sending host address matches a valid entry in an internal host binding table, queue the packet for local processing. The router updates this binding table with an internal algorithm.
- If the sending host address matches a valid entry in an internal NAT binding table, forward the packet to the corresponding LAN IP address. The router updates the NAT binding table with an internal algorithm.
- If the receiving port/protocol pair matches a reserved pair to support an internal process (for example, TCP Port 80 for the device web server process), queue the packet for local processing.
- If the receiving port/protocol pair matches a port forwarding rule, forward the packet to the LAN IP address according to that rule.
- If a DMZ host is configured, forward the packet to that LAN IP address.
- Queue the packet for internal processing.

DHCP Server

By default, the built-in DHCP server is enabled on the device. It assigns IP address, network mask, DNS server, and default gateway address to the DHCP clients on the LAN side. The default gateway and DNS server have the same IP address as the LAN side IP address of the router.

In the DHCP server configuration, you can select the range of client IP addresses to give out the Lease Time and the Local Domain Name. Furthermore, by using the DHCP reservation feature, you can reserve specific IP addresses for some devices with specific MAC addresses. With this, those devices can always be assigned the same IP addresses reserved for them each time they make a request to the DHCP server. See the [LAN Settings](#) and [DHCP Reservation](#) sections for more details.

Firewall

The firewall protects local processes and LAN side clients against certain basic threats from the WAN side (or the Internet), such as port scanning and a DOS (Denial of Service) attack. The firewall settings also allow you to selectively turn on or off the following related features:

- **NATRedirection** – If enabled, supports NAT Redirection, also known as NAT Loopback or Hairpin. The default is disabled.
- **DRDOSAttackProtection** – If enabled, protects against DOS attack if enabled. The default is disabled.
- **VPNPassThrough** – If disabled, blocks all VPN traffic if disabled. The default is enabled.

The settings of these features take effect only if the firewall is enabled. Otherwise, they take on their respective default values (that is, no **NATRedirection** or **DRDOSAttackProtection**; and **VPNPassThrough** is allowed).

Port Forwarding

As many as 20 port forwarding rules can be defined on the device. For each rule, a range of ports and a designated receiving LAN IP address must be specified such that incoming traffic arriving at any of those ports on the WAN side are forwarded to the same port at the designated IP address on the LAN side. You also can specify for each rule if it should only apply to packets transported over UDP, TCP, or both.

DMZ

The DMZ host in the router is the default LAN client address to which a packet received from the WAN side is forwarded when the router fails to find a matching LAN IP address or matching local process to forward the packet to. Note if the firewall is enabled, that the packet is still subject to firewall inspection before forwarding to the DMZ host.

QoS

QoS (Quality of Service) refers to the prioritization of network traffic based on the type of traffic. For example, time-critical traffic such as VoIP can be allocated the highest priority so it has a better chance of on time delivery to the destination. On the device, QoS policy applies to upstream traffic (LAN-to-WAN) only. Downstream QoS is entirely up to the ISP / upstream routers and switches. The upstream traffic is prioritized according to its type of service as indicated by the DiffServ/TOS bits in the IP header of each packet.

In the QoS settings, you can map the 64 possible types of service to one of the three priority classes: High, Medium, or Low. You also can specify the guaranteed minimum upstream bandwidth for each priority class. LAN side clients indicate the desired priority class of their outbound packets to the router by marking the DiffServ/TOS bits of their packets accordingly. See the [QoS Settings](#) section for more details.

In addition to the three priority classes, a fourth priority class known as the **Restricted** class is available. The Restricted class has the highest priority among the four classes. The guaranteed bandwidth for the Restricted class is allocated separately with its own parameter in the configuration.

Note that the total guaranteed bandwidth allocated to all the four priority classes is equal to the total available uplink bandwidth, which must be specified correctly in the **UpStreamBandwidth** parameter in the QoS settings for QoS to work properly.

VLAN Support in Router Mode

In router mode, the device can support VLAN (802.1Q) on the WAN side. When VLAN is enabled, incoming packets from the WAN side not belonging to the same VLAN are dropped, while all outgoing packets to the WAN side are tagged with the configured VLAN ID. The VLAN support is transparent to the devices on LAN side. The router removes the VLAN tag when forwarding packets to the LAN side.

LAN Switching Features

Instead of acting as a router, the device can be set to work as a 3-port switch. One of the ports is internal and is used by the device CPU only, while the two external ports (labeled as Internet and LAN) can be connected to other devices. This mode of operation is called bridge mode. In bridge mode, all the router features, such as DHCP server, firewall, and port forwarding, don't take effect. The QoS policy in this case is hardwired such that the native voice and related traffic always have the highest priority. This behavior isn't configurable. Furthermore, accessing the device management web pages from either of the two external ports is always allowed.

VLAN Support in Bridge Mode

When VLAN is enabled, packets sent to the device not belonging to the same VLAN are dropped, while packets sent by the device are tagged with the configured VLAN ID. The packets switched directly between the external ports, on the other hand, are not modified by the device.

Phone-to-Phone Collaborative Features (OBi504/508)

While each phone port functions independently of each other, an OBi5 Series device provides PBX-like features that tie the ports together to work as a phone system. The default Phone port digit map and call routing rules are set up such that you can:

- Dial *#n* to call phone port *n*, *n* = 1, 2, 3, ... 8 (Dialing *#n* from phone *n* is reserved for checking voicemail in phone *n*'s mailbox)
- Dial *#** to ring all the phones at once
- Dial *#0* to page out of the Line Out Port

Furthermore, when calling phone-to-phone with a prefix **96*, the caller can trigger the following behavior:

- If the called phone is idle (on-hook), it rings normally with a special Caller-ID that indicates the call is from the other Phone port.
- If the called phone is on-hook with a call on-hold, the calling phone picks up and resumes that call.
- If the called phone is ringing, the calling phone picks up and answers that call.
- For all other scenarios, the calling phone hears a busy tone.

You can also transfer an external call from Phone 1 to Phone 2 the usual way: while connected on an external call, hook flash and dial *#* to ring the other phone, then hang up to transfer when the caller phone rings or answers.

Star Codes

Star codes are short sequences of digits where each sequence serves as a command to the device to perform certain operation. Each sequence usually starts with the * key followed by a 2-digit code (such as *69).

You can use star codes to set the value of one or more configuration parameters. The device allows you to issue star code from the Phone port only. Every star code and its operation are defined with a short star code script parameter. The set of star codes that can be dialed from the Phone port is collectively referred to as a *star code profile*.

Controlling Calls Using Star Codes

The device has two star code profiles available in its configuration, known as Star Code Profile A and Star Code Profile B. Each profile has 30 star code script parameters, known as Code1 to Code30. You can select which star code profile to use by setting **Phone Port::StarCodeProfile** to A or B, or **None** if star codes aren't used.

A star code script is defined with the help of a number of predefined variables and actions. Each variable represents one or one group of configuration parameters. An action can be checking or setting the value of a variable, collecting a phone number, or calling a certain number.

Star Code List

Your device has the following star codes preprogrammed.

Preprogrammed Star Codes

Code	Description
*03	Request peer device to loop back media in the next outbound call
*04	Request peer device to loop back RTP packets in the next outbound call
*05	Tell device to periodically redial the last called number until the called party rings or answers
*06	Cancel the last repeat dial request
*07	Redial
*28	Make OBiBT Bluetooth Adapter discoverable for the next 120 seconds and set it as OBiBlueTooth 1 Note: You must attach one and only one OBiBT dongle to the unit when using this star code. Otherwise, the operation will fail.

Preprogrammed Star Codes (continued)

Code	Description
*29	Make OBiBT Bluetooth Adapter discoverable for the next 120 seconds and set it as OBiBlueTooth 2 Note: You must attach one and only one OBiBT dongle to the unit when using this star code. Otherwise, the operation will fail.
*56	Enable Call Waiting
*57	Disable Call Waiting
*60	Call Forward on Busy (Enter Number + #)
*61	Disable Call Forward in Busy
*62	Call Forward on No Answer (Enter Number + #)
*67	Block Caller ID (One Time)
*63	Disable Call Forward No Answer
*66	Repeat Dial
*68	Unblock Caller ID (One Time)
*69	Call Return
*72	Call Forward All (Enter Number + #)
*73	Disable Call Forward All
*74	Speed Dial Setup (Enter SD No. [1-99] then Tel No. + #) Note: Be careful with the Speed Dial Setup, as this will conflict with the Speed Dials set up on the OBiTALK portal. Speed Dials set up on the OBiTALK portal always overwrite anything set up via the device.
*75	Speed Dial Read-Back (Enter SD No.)
*76	Clear a Speed Dial
*77	Block Anonymous Calls
*78	Do Not Disturb – Turn On
*79	Do Not Disturb – Disable
*81	Block Caller ID (Persistent Mode)
*82	Unblock Caller ID (Persistent Mode)
*86	Disable Repeat Dial
*87	Unblock Anonymous Calls
*96,	Barge In
*98	Blind Transfer

Preprogrammed Star Codes (continued)

Code	Description
*4711	Use G711 Only on the next outbound call
*4729	Use G729 Only on the next outbound call

Star Code Script Variables

A star code script variable can be trunk-specific or Phone port-specific. The general format of a Phone port-specific variable is `$var` and it applies to the current Phone port where the star code is entered. The general format of a trunk-specific variable is `TK($var)`, where `TK` is the abbreviated name of a trunk (for example, SP1, SP2, or PP1). If `TK` isn't specified for a trunk-specific variable, it implies all the applicable trunks in the system. Variable names are case-insensitive.

The following `$var` variables are supported:

Supported \$var Variables

Variable	Meaning
<code>\$CFA</code>	Call forward unconditional enable (trunk-specific; admissible value: 0 for disable, 1 for enable)
<code>\$CFB</code>	Call forward busy enable (trunk-specific; admissible value: 0 for disable, 1 for enable)
<code>\$CFN</code>	Call forward no-answer enable (trunk-specific; admissible value: 0 for disable, 1 for enable)
<code>\$CFAN</code>	Call forward unconditional number (trunk-specific; admissible value: a token representing a call forward number)
<code>\$CFBN</code>	Call forward busy number (trunk-specific; admissible value: a token representing a call forward number)
<code>\$CFNN</code>	Call forward no-answer number (trunk-specific; admissible value: a token representing a call forward number)
<code>\$MWS</code>	Message waiting state (trunk-specific; admissible value: 0 for no new messages, 1 for one or more new messages)
<code>\$DND</code>	Do-not-disturb enable (trunk-specific; admissible value: 0 for disable, 1 for enable)
<code>\$BAC</code>	Block-anonymous caller enable (trunk-specific; admissible value: 0 for disable, 1 for enable)
<code>\$BCI</code>	Block outbound caller-ID enable (trunk-specific; admissible value: 0 for disable, 1 for enable)
<code>\$CWA</code>	Call-waiting enable (global; admissible value: 0 for disable, 1 for enable)
<code>\$BCI1</code>	Block caller-ID once (global; admissible value: 1 for enable)
<code>\$UBCI1</code>	Unblock caller-ID once (global; admissible value: 1 for enable)
<code>\$LBM1</code>	Loopback media (audio samples) once in the next call
<code>\$LBP1</code>	Loopback RTP packets once in the next call

Supported \$var Variables

Variable	Meaning
\$CDM1	Codecs to enable in the next call on this Phone port (temporarily overriding any codec preferences in device configuration) (port-specific; admissible value: An 8-bit unsigned number where each bit of its value represents one audio codec) • Bit0(LSB) = G711u • Bit1 = G711a • Bit2 = G726r16 • Bit3 = G726r24 • Bit4 = G726r32 • Bit5 = G726r40 • Bit6 = G729
\$LDN	Last dialed number (for redial) (global; read only)
\$BAR1	Enable Barge-In 1 on the next call (global; admissible value: 1 for enable)
\$Bxrn	Blind Transfer Target Number (global; admissible value: a token representing the target number)
\$LCR	Last caller's number on this Phone port (for call return) (global; read only)
\$SPD[n]	Number for the speed dial n ($n = 1$ through 99) (global; admissible value: literal or token representing a phone number)
\$CODE	One or more digits representing the variable part of a star code (see examples below; read-only)

Star Code Script Actions (ACT)

The general format of an action is `ACT(par, par, ...)`. You can set multiple variables with multiple `set()` actions with a single star code. Action names are case-insensitive.

The following actions are supported:

Star Code Script Actions (ACT)

Script	Actions
<code>set(VAR, token)</code>	Sets the given VAR to the value represented by token.
<code>call(token)</code>	Calls the number represented by token. PHONE Port::OutboundCallRoute is applied when making the call (but not the DigitMap).
<code>rpdi(token)</code>	Repeat-dials the number represented by token.
<code>coll(VAR)</code>	Collects a number from you and stores it as the value of one or more parameters represented by VAR. The number is collected with PHONE Port::DigitMap applied.
<code>say(token)</code>	Announces the value represented by token. Values are announced as a list of letters or numbers, where token can be a literal such as 1234, or another variable, such as \$CFAN or SP1 (\$CFBN)

Star Code Script Actions (ACT)

Script	Actions
<code>btdscvr(n)</code>	Makes the OBiBT dongle discoverable for the next 120 seconds. n = • 0 for BT1. • 1 for BT2.
<code>wifiap()</code>	Makes the device acts like a Wi-Fi Access Point when an OBiWiFi5G USB Dongle is attached.

Star Code Script Format

The general star code script format is `code, name, action1, action2, action3, ...`. Actions are carried out one-by-one in the order as specified in the script.

Restrictions:

- At most 1 `call` action per code.
- Either 1 `say` or 1 `call` action at most per code, and it must be the last action in the script.

Star Code Script Format

Script	Actions
<code>code</code>	The star code, such as <code>*72</code> . It can contain a variable part enclosed in parentheses, such as <code>*74(x xx)</code> . The variable parts you enter are stored in the variable <code>\$code</code> .
<code>name</code>	Descriptive name of the function of this star code, such as Call Forward Unconditional.
<code>action1, action2, ...</code>	Valid action with parameters

Star Code Script Examples

The following examples are taken from some of the default star code scripts in the device.

***69**, Call Return, `call($LCR)`

Calls the number of the caller who last rang the Phone port.

***07**, Redial, `call($Ldn)`

Redials the last dialed number.

***72**, Call Forward Unconditional, `coll($cfan), set($cfa,1)`

Collects a number from you according to the **DigitMap**, then sets the **CallForwardUnconditionalNumber** on all trunks to the collected value, and sets the **CallForwardUnconditionalEnable** on all trunks to **Yes**.

a To modify the script to enable **CallForwardUnconditional** on SP1 only, change it to:

***72**, Call Forward Unconditional SP1, `coll(SP1($cfan)), set(SP1($cfa),1)`

***67**, Block Caller ID Once, `set ($BCI1, 1)`

Enables masking of caller ID information once for the next call on any trunk.

***74(x|xx)**, Set Speed Dial, `coll ($Spd[$code])`

After you dial *74, the device expects one or two more digits from you, which represent a speed dial slot index (1 to 99). The 1- or 2-digit variable part is stored in the variable `$code`.

The device then plays a prompt tone and proceeds to collect a number from you according to the **DigitMap**. Finally the device stores the collected number in the given speed dial slot. If the slot already has a number specified, it is overwritten quietly with the new value.

***75(x|xx)**, Check Speed Dial, `say ($Spd[$code])`

After you dial *75, the device expects one or two more digits from you, which represent a speed dial slot index (1 to 99). The 1- or 2-digit variable part is stored in the variable `$code`.

The device then announces the number stored in the speed dial slot, or says "not available" if the slot is empty.

For more information on Star Code parameters, see the [Star Code Profile Parameter Guide](#) table in the [Parameter Reference](#) section.

Status Pages

The Status parameters pages show read-only values for certain parameters on your device.

System Status Page

The System Status page is divided into several sections:

- WAN Status
- WiFi Status
- Product Information
- OBiBluetooth Service Status
- SP Service Status
- OBiTALK Service Status

WAN Status Section

The status of the WAN (Ethernet) interface includes such information as the assigned IP address, default gateway, and subnet mask.

WiFi Status Section

This shows the status of OBiWiFi5G and includes such information as the assigned IP address, default gateway, and subnet mask.

Product Information Section

This status shows some basic product information, as well as the system up-time with the last reboot reason code in parentheses. The reboot reason codes are defined as follows.

Reboot Reason Codes

Reason Code	Description
0	Reboot on power cycle.
1	Operating system reboot.
3	Reboot after new profile invoked.

Reboot Reason Codes

Reason Code	Description
4	Reboot after parameter value change or firmware has changed and invoked via device web page.
5	Reboot after factory reset using the device hardware PIN.
6	New profile invoked AND profile URL changed.
7	Reboot from SIP Notify (Reserved).
8	Reboot from telephone port (IVR).
9	Reboot from webpage — no change in parameter values or firmware.
10	Reboot during OBiTALK signup.
12	Reboot after DHCP server offers IP, GW-IP, and/or netmask different from what the device is currently using.
13	Reboot on data networking link re-establishment.
18	Reboot on WAN IP address change.
19	Reboot on LAN IP address change.

OBiBT Bluetooth Service Status Section

This status shows the status regarding the OBiBT Bluetooth adapter attached to the unit and the corresponding OBiBlueTooth Services. The following status values are available:

- State can be one of the following values:
 - No Dongle: No authentic OBiBT dongle detected
 - Connecting <device-name>: Connecting to the named device
 - Disconnected: Not connected with any paired device
 - No Device To Connect: No device paired
 - Service Down: Connected to a mobile phone that doesn't have an active service (for example, no SIM card or no signal)
 - Service Disabled: OBiBlueTooth service has been disabled in the device configuration
 - Connected to <device-name>: Connected to the named device. OBiBlueTooth service is available only in this state
- Discoverable shows either "Not Discoverable" or the number of seconds for which OBiBT remains discoverable
- CallState can be one of the following values:
 - Incoming Call: BT detected an incoming call, but ringing has not started yet
 - Ring <Caller-ID>: Mobile phone is ringing. <Caller-ID> is the caller's number
 - Outgoing Call: Mobile phone is making a call that isn't answered yet
 - Voice Disconnected: Mobile phone is in a connected call, but the audio stays in the phone instead of going to OBiBT

- 0 Active Calls
- 1 Active Call: Mobile phone is in a connected call, with audio going to OBiBT
- BindingService shows the external paired device currently connected with the OBiBT

SPn Service Status Section

The *SPn* service status values indicate the current state of the service with regard to its configuration (or not) and if configured its registration status. If there are problems with the registration or authentication of the device with a prescribed service, the SIP 4xx error message is displayed here. This is very useful information for troubleshooting issues with SIP-based services.

OBiTALK Service Status Section

The status of the OBiTALK Service includes the following values:

- Status – Possible values are:
 - Normal (User Mode): The service is functioning normally.
 - Backing Off: The service is currently down, and the device is taking a short pause before retrying the connection.
- CallState – Possible values are:
 - *N* Active Calls, where *N* = 0, 1, ..., as many as the maximum number of calls allowed in the configuration.

LAN Status Page

The LAN Status page shows the devices currently on the LAN. There are two sections:-

- Attached Devices: All the devices that the router has discovered on the LAN side. Each entry has a MAC address and an IP address.
- DHCP Clients: All the DHCP clients that have an active lease with the DHCP server. Each entry has a Client Name, a MAC address, an IP address, and the lease expiration time (in seconds).

This page is available when the device is working in router mode.

Call Status Page

The Call Status page shows a number of running call statistics and state parameters for each active call currently in progress. For each entry on the call status page, the following buttons may be available:

- **Remove:** This button is available for all calls. Pressing this button ends that call.
- **Record:** This button is available for calls involving the Phone port only. Pressing this button allows you to record the current conversation in an audio (.au) file.

Call History Page

The Call History page shows the last 400 calls made with the device. Detailed call information is available, including what terminals were involved, the name (if available) of the Peer endpoints making the call and the direction / path the call took.

The Call History page also captures what time various events took place.

The Call History can be saved at any time by clicking on the “Save All” button. The Call History can be saved as an XML formatted file called `callhistory.xml`.

Services, Phone and Line Status (OBi312)

You can find statistics relevant to SP_n on the SP_n Stats page (where $n = 1-9$).

See the [Parameter Reference](#) for information on the parameters displayed on these pages.

Device Settings

You can control how devices perform Bluetooth settings, speed dialing services, and user-defined digit maps. You can also control device codec features, handset tones, and ringtones.

OBiBlueTooth Settings

The OBiBlueTooth feature requires an OBiBT Bluetooth adapter connected to the USB port of the device. OBiBT is compatible with Bluetooth 1.1 to 4.0 and supports the HFP (handsfree) profile. When paired with a mobile phone, OBiBT plays a role similar to a Bluetooth headset from the perspective of the mobile phone.

Up to two OBiBT Bluetooth adapters can be attached to the device (requires an external USB hub if more than one). Each dongle must be set up to associate with either OBiBlueTooth 1 (BT1) Service or OBiBlueTooth 2 (BT2) Service.

OBiBlueTooth Call Features

OBiBlueTooth is the voice service that is enabled with an attached OBiBT dongle paired with an in-service mobile device. This service allows you to make or receive mobile calls using a conventional phone attached to the device Phone port, or by bridging in a VoIP call over an SP service or OBiTALK service. OBiBlueTooth only allows one incoming or outgoing call on OBiBT at a time. In other words, there's no 3-way call or call-waiting support on OBiBlueTooth. The supported call features are similar to the equivalent ones supported under SP or OBiTALK service.

Phone Interface Features

This section provides information on phone interface features such as repeat dialing, audio line in, and audio line out.

Repeat Dialing Service

Repeat dialing service is when a user dials ***05** to tell the device to redial the last called number repeatedly while the phone is on-hook until the called party rings or answers. When that happens, the device rings the Phone port and you can pick it up to talk to the called party. Typically the last called number was busy when you invoke this feature, but the device allows this feature for all cases.

This feature can be controlled with the following parameters (under the ***Phone Port – Calling Feature*** section):

- **RepeatDialInterval** = the minimum number of seconds between each redial. Default is 30 seconds.

- **RepeatDialExpires** = the maximum duration in seconds when the repeat dialing remains active. Default is 1800 seconds.

Dial ***06** to cancel Repeat Dialing. Only one repeat dial request is supported. Dialing ***05** while a repeat dial is in progress is rejected with a fast busy tone. If ***05** is accepted, the device plays normal dial tone.

Notes:

- The first redial happens 5 seconds after the phone is on-hook following ***05**.
- When the phone is off-hook or rings for an incoming call, the device pauses redial and cancels the call if it's already dialed but the peer device isn't ringing yet.
- As soon as the phone goes on hook or ringing stops without any calls on hold, repeat dialing resumes in 5 seconds.
- If the called party answers before the local caller, the device sends a normal ringback tone over RTP to the called party.
- The ring for alerting local user when the called party rings or answers is taken from outgoing trunk's **RepeatDialRing** parameter.
- Repeat Dial service can't be used on the LINE port. If the last call was made over the LINE port, dialing ***05** results in fast busy.
- Repeat Dial calls aren't logged to call history, except the last and successful one when the called party rings or answers.

For more information on Phone port parameters, see the [Phone Port Status Parameter Guide](#) table in the [Parameter Reference](#) section.

Audio Line In

OBI devices have an Audio Line In port that you can connect to an external audio source with a 3.5 mm stereo jack. It takes a stereo source and converts it into a mono signal internally and samples it at 8 kHz. This audio source can be used as for Music On Hold (MOH).

This port is referenced with the short name **AI** (case-insensitive) in the configuration. To use it for MOH when a phone user places the call on hold, you can specify the **MOHServiceNumber** phone port parameter = `ai(0)`.

You can route a call to ring `ai(0)` to let the caller listen to the audio coming from the Audio Line In port. **AI** can answer as many as four calls simultaneously.

Audio Line Out

OBI504/508 devices have an Audio Line Out port that you can connect to an external PA system for paging with a 3.5 mm stereo jack. The same audio signal is played on both left and right channels, sampled at 8 kHz. The audio port is referred to with the short name **AO** (case insensitive) in the configuration. By default, a phone user can dial **#0** to page out of the AO port, with the following rule in the outbound call route: `{#0:ao}`. A similar routing rule may be added elsewhere in the configuration to route a call to send audio out at the AO port. The AO port can only take one call at a time. The OBI also plays a short beep prior to playing caller's voice at the AO port for every call.

The Audio Line In source is looped back to the Audio Out port in the analog domain, and therefore isn't subject to quality loss due to ADC/DAC. When paging starts, the device automatically lowers the level of the

analog Audio Line In signal in favor of the caller's audio signal. When paging ends, the device gradually raises the level of the analog Audio Line In signal to full volume.

Codec Profile Features

Two codec profiles are available on the devices, selectable per trunk (SP n , OBiTALK, OBiBlueTooth). To select a codec as the preferred codec in this profile, set the priority of that codec to be highest among all the enabled codecs in this profile. Each of the SP and OBi services can be assigned a codec profile in its corresponding configuration. The codec list to use when setting up a call on the underlying service is formed from the list of enabled codecs in the chosen profile and ordered according to the assigned priorities in the profile.

For more information on codec profile parameters, see the [Codec Profile X Web Page \(X = A, B\)](#) table in the [Parameter Reference](#) section.

Tone and Ring Patterns

Your device enables you to create customized tone patterns and ringtones.



Tone and Ring Profile A default settings are set for North American telephone standards. Tone and Ring Profile B default settings are set for Australian telephone standards. Tone profiles for other countries are available for download from the OBiTALK forum.

Tone Profile Features

The general format for tone profiles is: `[field-1];[field-2];[field-3];...;[field - 6]`

Use a semicolon to separate the configuration fields.

Note that a tone profile pattern must contain no spaces.

Field-1 Composition

This field describes frequency components used for tone synthesis and it supports as many as three different frequencies.

The frequency expression is a string of numeric values with the notation '+' or '-'. The numeric values are the frequency's decimal values in Hz and amplitude in dBm (maximum 3 dBm). Different frequencies are separated by a comma.

Example: 350-18,440-18,550+2

This example means:

- The first frequency at 350 Hz with strength at -18 dBm
- The second frequency at 440 Hz with strength at -18 dBm
- The third frequency at 550 Hz with strength at +2 dBm

Field-2 Composition

This field describes the overall tone playback duration in seconds.

The expression is a numeric value, and supports as many as 3 decimated digits. The numeric value can be negative, zero, positive, or skipped:

- Negative value: tone plays indefinitely
- Zero value: tone playback is skipped
- Positive value: Normal playback duration
- No value: tone plays indefinitely

Example: 30.234

This example means:

- Tone playback terminates after 30.234 seconds

Field-3 to Field-6 Composition

Field - 3/4/5/6 share the same definition, and each field describes one single cadence segment. Together, the four fields form a macro-segment, which repeats until tone playback expires.

The expression is a string of numeric values with the special notation '/', '(', ')' and ','.

Its format is: `t(f_0/on_0+off_0,f_1/on_1+off_1,f_2/on_2+off_2,f_3/on_3+off_3)`

- `t`: the cadence segment duration in seconds
 - Negative value: tone plays indefinitely
 - No value: tone plays indefinitely
 - Zero value: the duration of this particular segment is zero
 - Positive value: Normal playback duration
- `f_0/1/2/3`: a digit to describe which frequency components are used for the synthesis, and can be one of following 8 options (0 through 7)
 - 0: No frequency specified (silent tone)
 - 1: The first frequency
 - 2: The second frequency
 - 3: The first and second frequencies
 - 4: The third frequency
 - 5: The first and third frequencies
 - 6: The second and third frequencies
 - 7: The first and second frequencies if two or more than two frequency components, or the first frequency if only one frequency component is available.

If no value is provided for `f_0/1/2/3`, it automatically uses the combination of the first one or two available frequency components.

- `on_0/1/2/3`: the tone active time in seconds
 - Negative value: Not allowed
 - No value: infinite tone active time
 - Others: normal tone active time (as many as 3 decimated digits)
- `off_0/1/2/3`: the tone inactive time in seconds
 - Negative value: Not allowed

- No value: infinite tone inactive time
- Others: normal tone inactive time (as many as 3 decimated digits)

Example: 4 (1/.3+2.34,3/2+1.5)

This example means:

- Use the first frequency to generate a tone for 0.3 seconds
- Follow this tone with 2.34 seconds of silence
- Use a combination of the first and second frequencies to generate a tone for 2 seconds
- Follow this tone with 1.5 seconds of silence
- The cadence operates repeatedly for 4 seconds.

Tone Examples

These examples show the interpretation of a few common tone patterns:

Dial Tone

DIAL, "350-18,440-18"

Dial tone is generated as a mixture of two frequency components:

350 Hz at -18 dBm and 440 Hz at -18 dBm

The expiration time is infinite, and tone active time is infinite.

Busy Tone

BUSY, "480-18,620-18;10;(.5+.5)"

Busy tone is generated as a mixture of two frequency components:

480 Hz at -18 dBm and 620 Hz at -18 dBm

The expiration time is exactly 10 seconds. It has only one cadence segment, which has tone active 0.5 second and tone inactive 0.5 second.

Prompt Tone

PROMPT, "480-16;10"

Prompt tone is generated from a single frequency component:

480 Hz at -16 dBm. The expiration time is exactly 10 seconds. It has only one cadence segment, which has tone infinite active time.

SIT Tone

SIT_1, "985-16,1428-16,1777-16;20;(1/.380+0,2/.380+0,4/.380+0,0/0+4)"

Special information tone (SIT) is generated from a set of frequency components:

- First frequency: 985 Hz at -16 dBm
- Second frequency: 1428 Hz at -16 dBm
- Third frequency: 1777 Hz at -16 dBm

The expiration time is exactly 20 seconds. It has only one cadence segment, which includes 4 on-off sections. The segment has infinite repeating time:

- The first on-off section: generated by the first frequency component, and it has 0.38 tone second active time and 0 inactive time.
- The second on-off section: generated by the second frequency component, and it has 0.38 tone second active time and 0 inactive time.
- The third on-off section: generated by the third frequency component, and it has 0.38 tone second active time and 0 inactive time.
- The fourth on-off section: only generate silence since no frequency component is specified. It has tone 0 second active time and 4 seconds inactive time.

Stutter Tone

```
STUTTER, "350-18,440-18;20;.2(.1+.1);()"
```

Stutter dial tone is generated from a mixture of two frequency components:

350 Hz at –18 dBm and 440 Hz at –18 dBm. The expiration time for the entire tone is exactly 20 seconds. It has two cadence segments.

- The first segment includes only one on-off section, on 0.1 second and off 0.1 second, and on-off repeats for 2 seconds.
- The second segment includes one on-off section, and has infinite repeating time and infinite tone active time, and plays until the entire tone duration has elapsed.

For more information on Tone Profile A & B parameters, see the [Tone Profile A & B Parameter Guide](#) table in the [Parameter Reference](#) section.

Ring Patterns

The general format of an OBi Ring Profile is as follows: [field-1];[field-2];...;[field - 5]

Use a semicolon to separate as many as five configuration fields.

Note that a tone profile pattern must contain no spaces.

Field-1 Composition

Field-1 describes the overall ringing duration in seconds.

The expression is a numeric value, and supports as many as 3 decimated digits.

The numeric value can negative, zero, and positive:

- Negative value: Ringing lasts indefinitely
- No value: Ringing lasts infinitely
- Zero value: Ringing is skipped
- Positive value: Normal ringing duration

Example: 30.5

This example illustrates a ringing tone that terminates after 30.5 seconds.

Field-2 to Field-5 Composition

Fields-2/3/4/5 share the same definition, and each field describes one single cadence segment. Together, the four fields form a macro-segment, which repeats until ringing expires.

The expression is a string of numeric values with the special notation '(', ')', and ','.

It has the format as per the following construct:

`t(on_0+off_0,on_1+off_1,on_2+off_2,on_3+off_3)`

`t`: The cadence segment duration in seconds.

- Negative value: Ringing indefinitely
- No value: Ringing indefinitely
- Zero value: Ringing is skipped
- Positive value: Normal ringing duration

`on_0/1/2/3`: The ring active time in seconds.

- Negative value: Not allowed
- No value: Infinite ring active time
- Others: Normal ring active time (as many as 3 decimated digits)

`off_0/1/2/3`: The ring inactive time in seconds

- Negative value: Not allowed
- No value: Infinite ring inactive time
- Others: Normal ring inactive time (as many as 3 decimated digits)

Example: `4(.3+2.34,2+1.5)`

This example illustrates a ringing tone comprised of two segments. Ringing is active for 0.3 seconds, followed by 2.34 seconds of silence, then ringing for 2 seconds, and followed by 1.5 seconds of silence.

This cadence operates repeatedly for 4 seconds.

For more information on call waiting parameters, see the [Tone Profile A & B Parameter Guide](#) table in the [Parameter Reference](#) section.

For more information on ring profile parameters, see the [Ring Profile A & B Parameter Guide](#) table in the [Parameter Reference](#) section.

User Settings

Use the device web page to configure user speed dial numbers and user-defined digit maps.

Speed Dial Numbers

Each OBi device supports 99 speed dial numbers. The 99 speed dial slots are numbered from 1 to 99 and are invoked by dialing a 1- or 2-digit number corresponding to the slot number.

Speed dial values can be set using the configuration web page, remote provisioning, or star code (see the [Star Codes](#) section in this guide for more details). The value can be a number just like the one you normally dial, with or without any service access code prefix, such as `**9200112233`, `**214089991123`, `4280913`, and so forth. It may also include explicit trunk information with the general format `TK(number)`,

where TK= SPn (n=1-8), BT1, BT2, or PP. For example, PP (ob200112233), SP2 (14089991123), BT2 (4280913), and so forth.

If trunk information isn't specified in the speed dial entry, the device applies **DigitMap** and **OutboundCallRoute** when making the call. Otherwise, neither **DigitMap** nor **OutboundCallRoute** is applied.

Using Speed Dial Number as Ad Hoc Gateway

If an external gateway doesn't require authentication, its access number can be stored in one of the 99 speed dial slots to allow ad hoc direct dialed gateway calls. To do this, dial the gateway's speed dial, followed by a *, followed by the target number. That is <gateway-speeddial> * <target-number>. For example, the gateway access number pp (ob200333456) is stored at speed dial 8, and you can dial 8*14085551234 to call 14085551234 using the given gateway.

Note: Only gateways that are accessed with an OBi number can be used this way.

For more information on speed dial parameters, see the [Speed Dial Parameter Guide](#) table in the [Parameter Reference](#) section.

User-Defined Digit Maps

For an explanation of this feature, please refer to the [User-Defined Digit Maps](#) section in the [Digit Map Configuration](#) section.

For more information on user-defined digit map parameters, see the [User-Defined Digit Maps Parameter Guide](#) table in the [Parameter Reference](#) section.

Network Settings

You can configure the network options through the WAN and LAN settings submenu.

WAN (Internet) Settings

For more information on WAN settings parameters, see the [WAN Settings Parameter Guide](#) table in the [Parameter Reference](#) section.

LAN Settings

For more information on LAN settings parameters, see the [LAN Settings Parameter Guide](#) table in the [Parameter Reference](#) section.

DHCP Reservation

You can reserve as many as 20 specific IP addresses for the DHCP server to give out to DHCP clients with specific MAC addresses.

For more information on DHCP parameters, see the [DHCP Reservation Parameter Guide](#) table in the [Parameter Reference](#) section.

Firewall and DMZ

Firewall and DMZ apply only when the device is set to work in the router mode. The firewall filters incoming packets from the WAN side only. It provides protection against some threats from the WAN side. A global firewall enable option allows disabling all the firewall components. In addition, the three firewall-related features, **NATRedirection**, **DRDOSAttackProtection**, and **VPNPassThrough**, take effect only if the firewall is enabled.

For more information on Firewall and DMZ parameters, see the [Firewall and DMZ Parameter Guide](#) table in the [Parameter Reference](#) section.

Port Forwarding

A port forwarding rule is useful for supporting a server application on a LAN client, such as FTP Server or HTTP Server. You can define as many as 20 port forwarding rules.

For more information on port forwarding sharing parameters, see the [Port Forwarding Parameter Guide](#) table in the [Parameter Reference](#) section.

QoS Settings

QoS only applies to upstream traffic to the WAN side. The QoS settings described here take effect in router mode only. It is based on Traffic Control (TC) and Hierarchy Token Bucket (HTB). Refer to the HTB home pages for details on TC and HTB (<http://luxik.cdi.cz/~devik/qos/htb/>).

Four priority classes of upstream traffic are defined in the device QoS policy: The Restricted class has the highest priority, followed by the High, the Medium, and the Low classes. To fully utilize the upstream bandwidth, it is important to have a relatively accurate upstream bandwidth estimation configured in the **UpStreamBandwidth** parameter. This tells the device the total upstream bandwidth to allocate to the four priority classes.

The guaranteed uplink bandwidth for Restricted class traffic is configured in the **RestrictedBandwidth** parameter. The rest of the upstream bandwidth (the **UpStreamBandwidth** less the **RestrictedBandwidth**) is divided among the High, Medium, and Low classes proportionally according to the assigned weighting factor for each class in the configuration. Let W_{high} , W_{medium} , and W_{low} be the respective weights assigned to the High, Medium, and Low classes, the corresponding guaranteed uplink bandwidth for each class is calculated using the following formulas:

- $BW_{\text{high}} = (\text{UpstreamBandwidth} - \text{RestrictedBandwidth}) * W_{\text{high}} / (W_{\text{high}} + W_{\text{medium}} + W_{\text{low}})$ (kbps)
- $BW_{\text{medium}} = (\text{UpstreamBandwidth} - \text{RestrictedBandwidth}) * W_{\text{medium}} / (W_{\text{high}} + W_{\text{medium}} + W_{\text{low}})$ (kbps)
- $BW_{\text{low}} = (\text{UpstreamBandwidth} - \text{RestrictedBandwidth}) * W_{\text{low}} / (W_{\text{high}} + W_{\text{medium}} + W_{\text{low}})$ (kbps)

The native voice-related traffic is always classified as Restricted. Other network packets are classified based on the Differentiated Service Code Point (DSCP) in their IP headers. The 64 possible DSCP codes (0 – 63) can be mapped into one of the four priority classes using the configurable DSCP to Priority Class Mapping table.

For more information on QoS parameters, see the [QoS Parameter Guide](#) table in the [Parameter Reference](#) section.

Polycom Wi-Fi Wireless Network Adapter

Wi-Fi is supported by OBi 300/302/312 only. These devices support the 802.11 b/g/n wireless standards so that you can use a Polycom Wi-Fi wireless network adapter with the USB 2.0 port of the devices.

From an IP routing point of view, the Polycom Wi-Fi wireless network adapter is an additional WAN interface. If both WAN interfaces are connected (Ethernet port and OBiWi-Fi), the traffic destined to the WAN side routes through the Ethernet interface only, unless a) the WAN (Ethernet) interface and OBiWi-Fi are on different subnets and b) the destination address is on the same subnet as the Polycom Wi-Fi wireless network adapter.

If you set the device to function as a LAN switch rather than a router, the Polycom Wi-Fi wireless network adapter is disabled internally.

WiFi Scan

The WiFi Scan device page offers a familiar user interface to let you scan for access points in the neighborhood.

To automatically connect to WiFi access point:

- 1 Click on the page of the available AP to connect to.
If the AP requires authentication but the device does not have any valid credential, a page is returned to prompt you to enter a password or pass-phrase
- 2 Press **Connect** to continue.

To manually connect to WiFi access point:

If your AP doesn't show up as a listed device on this page, perhaps its SSID isn't broadcasted.

- Click **Add a Network**.
Enter the SSID and security credentials manually. The **Manage Networks** link takes you back to the **WiFi Settings** device page. The **Scan For Networks** link reloads this page in order to rescan for the access points in the neighborhood.

Network Settings

See the [WAN \(Internet\) Settings](#) section.

For more information on network settings parameters, see the [WAN Settings Parameter Guide](#) and [LAN Settings Parameter Guide](#) tables in the [Parameter Reference](#) section.

Call Routing

Call routing is the process by which the device sets up a call bridge or a (endpoint) call based on such information as the trunk on which the call originates, the caller's number, the called number, and so forth. Call routing rules are parameters used to instruct the device how to route calls. A call can transform into a call bridge or an endpoint call after being routed by the device according to the given routing rules.

Every call has to be originated from somewhere. From the device's perspective, calls originated from the trunk side are considered inbound calls, while calls originated from an endpoint are outbound calls. The call routing rule syntaxes for inbound calls and outbound calls are slightly different, and are explained in the following section.

Call routing rule configuration relies heavily on digit maps. If you are not familiar with how digit maps work, please read the [Digit Map Configuration](#) section in this guide.

Inbound Call Route Configuration

Every trunk has a corresponding **InboundCallRoute** parameter in the device configuration. It is a comma-separated list of rules where each rule is also surrounded by a pair of curly braces `{ }`. No extra white spaces are allowed. These rules tell the device how to handle an inbound call, such as sending it to the Phone port (and ringing the attached phone(s)), sending it to the Auto Attendant for further routing (interactively with the caller), or making another call on a specific trunk to bridge with this call.

The general format is:

InboundCallRoute:= rule OR {rule},{rule},...

Note that the curly braces can be omitted if there is only one rule in the route. The OR operator isn't part of the parameter syntax. It is used here to separate alternative values only.

A rule has the following format:

rule := peering-list : terminal-list

The following table shows the rule formats.

Rule Formats

Rule	Format	Notes
peering-list :	peering,peering,...	Comma-separated list of 0 or more peering objects.
terminal-list :	terminal,terminal,...	Comma-separated list of 0 or more terminal objects.
peering :	caller-list > callee-list	

Rule Formats

Rule	Format	Notes
caller-list :	caller caller caller ...	Vertical bar-separated list of 0 or more caller objects.
callee-list :	callee callee callee ...	Vertical bar-separated list of 0 or more callee objects.
caller :	number OR embedded-digit-map OR ? OR @	? = anonymous, @ = any number but anonymous.
callee :	number OR embedded-digit-map OR @	
terminal :	PHx OR AAx OR LIx (arg) OR SPx (arg) OR PPx (arg)	arg object is optional.
arg :	cid > target	
x :	1 OR 2 OR 3...	Where applicable. Can be omitted if x = 1.
cid :	spoofed-caller-number OR \$1	
target :	number-to-call OR \$2	
embedded-digit-map :	(Mlabel) OR digit-map	

Notes:

- **Terminal-list** can be empty, which means to block this call. The preceding ‘.’ can’t be omitted. As many as four terminals can be specified in the list. The listed terminals are called/rung by the device simultaneously. This operation is known as forking the call. A terminal can be a trunk or an endpoint.
- Abbreviated terminal names are case-insensitive.
- **Number** and **number-to-call** are literal strings, such as 14089991234.
- **Digit-map** is just any proper digit map, such as (1xxx|xx.). Make sure to include the enclosing parentheses.
- **Spoofed-caller-number** is a literal string, such as 14081112233, to be used as the caller number for making a new call on the specified trunk.
- (Mlabel) is a named digit map, where label is the abbreviated name of any terminal that has a digit map defined: SP1, SP2, SP3, SP4, LI1, PP, PH, PH2, or AA.
- \$1 is an internal variable containing the value of the caller number of this inbound call, after any digit map transformation in the matched caller object of the matched peering object in the peering-list.
- \$2 is an internal variable containing the called number of this inbound call, after any digit map transformation in the matched callee object of the matched peering object in the peering-list.

More notes on peering-list and peering objects:

- **Peering-list** is optional in **InboundCallRoute**. If the peering-list is empty, the succeeding ‘.’ can be omitted also. An empty peering-list implies a single peering object whose caller object list matches any caller number. That is, the following **InboundCallRoutes** are all equivalent:

➤ ph

- {ph}
- {:ph}
- {?|@>@:ph}
- Callee-list in a peering object can be empty. It implies the callee object @, meaning any called number. The preceding '>' can be omitted if callee-list is empty.
- Caller-list in a peering object can be empty. It implies the caller-list @|?, meaning any caller number including anonymous. The succeeding '>' can't be omitted if caller-list is empty but not the callee-list.

More notes on the `arg`, `cid`, and `target` objects:

- The `cid` object inside an `arg` object is optional. If omitted, it implies no caller-ID spoofing when making the call on the specified trunk. The succeeding '>' can be omitted if `cid` is omitted.
- The `target` object inside an `arg` object is optional. If omitted, it implies the target \$2, which means to call the original called number after applying any necessary digit map transformation implied by the rule. The preceding '>' can't be omitted if `target` is omitted but `cid` isn't.
- `arg` object is optional. If omitted, it implies the `arg` with the target \$2 and no `cid`. If `arg` is omitted, the succeeding parentheses () can be omitted also.

An inbound call matches a rule if its caller-number/callee-number matches one of the peering objects of the rule. Peering objects are tested in the order left and right, and the first matched peering object wins. Rules are also checked in the order left to right, and the first matched rule wins. Therefore it is important that you place the more specific rules first in the **InboundCallRoute** if multiple rules can potentially match the same inbound call.

InboundCallRoute Examples

ph OR {ph} OR {:ph} OR {@|?>@:ph} (all equivalent)

It says: Ring the Phone port for all incoming calls. This is the default **InboundCallRoute** for all trunks.

```
{(14081223330|15103313456):aa},{(1800xx.|1888xx.):}, {ph}
```

It says: Ring both Phone port and AA for calls coming from 1 408 122 3330 or 1 510 331 3456, block all 800, 888, and anonymous calls, and ring the Phone port for all other calls.

```
{(x.4081113333|x.4152224444):aa},{ph}
```

It says: Ring the AA for calls coming from any number that ends with 408 111 3333 or 415 222 4444, and ring the Phone port for all other calls. Be sure to include the enclosing parentheses in this example, since "x." is a digit map specific syntax.

```
{200123456:aa},{sp1(14083335678)}
```

It says: Ring the AA for calls coming from 200123456. For all any other call, bridge it by calling 1 408 333 5678 using SP1 Service.

Outbound Call Route Configuration

Every endpoint has an **OutboundCallRoute** parameter in the device configuration. It tells the device where to send the call when the endpoint attempts to make a call. Endpoints can call each other or an outside number using one of the trunks. The **OutboundCallRoute** syntaxes are almost identical to those of the **InboundCallRoute**. The differences are mainly in the implied value when an optional field is omitted, no

caller objects, and one and only one terminal object per terminal-list in an **OutboundCallRoute**. Forking isn't supported when routing outbound calls.

The general format is:

OutboundCallRoute := rule OR {rule}, {rule},

Note that the curly braces can be omitted if there is only one rule in the route. The OR operator isn't part of the parameter syntax. It is used here to separate alternative values only.

A rule has the following format:

```
rule := callee-list : terminal
```

where

- `callee-list` := `callee|callee|callee| ...`(vertical bar separated list of 0 or more callee object)
- `callee` := `number` OR `embedded-digit-map` OR `@` (`@` = any number)
- `terminal` := `PHx` OR `AAx` OR `LIx(arg)` OR `SPx(arg)` OR `PPx(arg)` (`arg` object is optional)
- `arg` := `cid > target`
- `x` := 1 OR 2 OR 3...(where applicable. Can be omitted `x` = 1.)
- `cid` = spoofed-caller-number
- `target` = `number-to-call` OR `$2`
- `embedded-digit-map` = `(Mlabel)` OR `digit-map`

Notes:

- A terminal can be a trunk or another endpoint.
- Abbreviated terminal names are case-insensitive.
- Number and `number-to-call` are literal strings, such as 14089991234.
- Digit-map is just any proper digit map, such as `(1xxx|xx.)`. Make sure to include the enclosing parentheses.
- Spoofed-caller-number is a literal string, such as 14081112233, to be used as the caller number for making a new call on the specified trunk.
- `(Mlabel)` is a named digit map where label is the abbreviated name of any terminal that has a digit map defined: SP1, SP2, LI, PP, PH, or AA.
- `$2` is an internal variable containing the called number of this outbound call, after any digit map transformation in the matched callee object.
- `Callee-list` can be empty, which implies the single callee object `@`, which means any called number. The succeeding `:` can be omitted also when `callee-list` is empty.

More notes on the `arg`, `cid`, and `target` objects:

- The `cid` object inside an `arg` object is optional. If omitted, it implies no caller-ID spoofing when making the call on the specified trunk. The succeeding `>` can be omitted if `cid` is omitted.
- The `target` object inside an `arg` object is optional. If omitted, it implies the target `$2`, which means to call the original called number after applying any necessary digit map transformation implied by the rule. The preceding `>` can't be omitted if `target` is omitted but not the `cid`.
- `arg` object is optional. If omitted, it implies the `arg` with the target `$2` and no `cid`.

An outbound call matches a rule if its called number matches one of the callee objects of the rule. Callee objects are tested in the order left to right, and the first matched callee wins. Rules are also checked in the

order left to right, and the first matched rule wins. Therefore it is important to place the more specific rules first in the **OutboundCallRoute** if multiple rules can potentially match the same outbound call.

Note that every endpoint also has a digit map defined. The user-dialed number is completely processed with the endpoint's digit map first before it is passed to the **OutboundCallRoute** for a routing decision. Therefore the number used for matching call routing rules has already incurred the transformations, if any, implied by the digit map. Remember this fact when crafting your own **OutboundCallRoute**.

OutboundCallRoute Examples

`sp1 OR {SP1} OR { :SP1 } OR { @:Sp1 }` (all equivalent)

This rule says: Make all calls using the SP1 Service, without any caller-id spoofing or digit transformation.

```
{ (<#: >|911):li }, { **0:aa }, { ***:aa2 }, { (Mpli):pli }, { (<**1:> (Msp1)):sp1 }, { (<**2:> (Msp2)):sp2 }, { (<**8:> (Mli)):li }, { (<**9:> (Mpp)):pp }
```

This is the default **OutboundCallRoute** for the Phone port. It says:

- Dial # key to connect to the LINE port. The device takes the LINE port hardware off-hook but won't dial out any digit. The net result is that you hear a dial tone generated by the PSTN company if a working line is connected to the LINE port. You can then continue to dial the PSTN number directly to the phone company. Note that from the device's point of view, the PSTN call is connected the moment it takes the LINE port to the off-hook state.
- Use the PSTN Line to call 911.
- Dial **0 to invoke AA1 or AA.
- Dial *** to invoke the local device configuration IVR (a.k.a AA2).
- (Mpli) and pli are substituted with the PrimaryLine's abbreviated name.
- Use SP1 Service to call all numbers that start with **1 and subsequent digits matching SP1 Service's **DigitMap**. Remove the **1 prefix from the resulting number before making the call.
- Use SP2 Service to call all numbers that start with **2 and subsequent digits matching SP2 Service's **DigitMap**. Remove the **2 prefix from the resulting number before making the call.
- Use the LINE port to call all numbers that start with **8 and subsequent digits matching LINE port's **DigitMap**. Remove the **8 prefix from the resulting number before making the call.
- Use the OBiTALK Service to call all numbers that start with **9 and subsequent digits matching OBiTALK Service's **DigitMap**. Remove the **9 prefix from the resulting number before making the call.

Digit Map Configuration

Digit Map Rules and Elements

A digit map serves to transform and restrict the number that can be dialed or called, and determine if you dialed sufficient digits to form a complete number. Each map is composed of one or more rules surrounded by parentheses (which MUST NOT be omitted). Here is the general format of a digit map:

```
(rule|rule|...|rule)
```

A digit map rule is a rule for matching a given sequence of digits. It can contain extra white spaces for readability. All spaces are removed by the device during parsing. A rule can contain one or more of the following elements:

- **literals** – Any combination of 0-9, *, #, +, -, A-Z, a-z, except m, M, s, S, x, X, which have special meaning in the digit map syntax. It matches digit sequences with exactly the same literals.
- **'literals'** – Everything inside a pair of single quotes is treated as a literal except for the single quote (') character.
- **x** – a wild card digit that matches any digit from 0-9. **x** is case-sensitive.
- **x.** – matches 0 or more **x**.
- **[123-7]** or **[135]** – A set of 1 or more digits surrounded by pair of []. It matches any digit in the set. The **-** syntax represents an inclusive digit range, such as 0-9, 3-7. So **[123-7]** is equivalent to **[1-7]** or **[1234567]**.
- **S, S0, S1, S2, ...S9** – Digit timer of 0, 1, 2, ...,9 seconds. **S** is equivalent to **S1**. **S0** is the same as "blank". You can concatenate multiple **S** elements together if you need more than 9 seconds timeout, such as **S9S5** for a 14-second timeout. **S** is case-sensitive. It should only be used either as the first element of a rule for hot/warm line implementation, or as the last element of a rule as a means of overriding the default interdigit timer.
- **<elements:literals>** – Substitute the digit sequence matching elements with the given literals. Single quote syntax isn't needed or allowed for the literals in this context. Special characters can be used here as they don't apply in this context either. Elements can be empty, in which case the ':' can be omitted. This case is useful for inserting some extra digits in certain part of the dialed digits. The literals part can be empty also but the ':' MUST NOT be omitted. This case is useful for removing part of dialed digits. Elements and literals MUST NOT be both empty.
- **(map)** – An embedded digit map for matching subsequent digits.
- **(Mlabel)** – A named embedded digit map for matching subsequent digits, where label is one of abbreviated terminal names. Possible choices are:
 - **(Msp1)** for **SP1 Service::DigitMap**
 - **(Msp2)** for **SP2 Service::DigitMap**
 - **(Msp3)** for **SP3 Service::DigitMap**
 - **(Msp4)** for **SP4 Service::DigitMap**
 - **(Mpp)** for **OBiTALK Service::DigitMap**
 - **(Mli)** for **LINE Port1::DigitMap**
 - **(Mph)** for **PHONE Port::DigitMap**
 - **(Mph2)** for **PHONE2 Port::DigitMap**
 - **(Maa)** for **Auto Attendant::DigitMap**
 - **(MtgN)** for **TrunkGroupN::DigitMap**, **N=1,2,3,4**
 - **(MvgN)** for **VoiceGatewayN::DigitMap**, **N=1,2,3,4,5,6,7,8**

Starting with release 1.2, the following elements are added:

- **X** – A wildcard digit that matches 0-9 or *. This is equivalent to **[x*]** or **[0-9*x]**
- **@** – A wildcard character that matches any alphanumeric character except #
- **x?** – matches 0 or 1 **x**
- **@?** – matches 0 or 1 **@**

- `[^...]` – matches any single alphanumeric character that isn't in the set
- Allow alphanumeric and wildcard inside a set `[ ]`, such as `[x]`, `[X#]`, `[@#]`, `[a-zA-Zx]`

The last two elements imply that the device digit maps are recursive. Recursive digit maps allow digit maps to be re-used and make their specification more compact and readable. It is important that you don't specify digit maps that lead to infinite recursion. For example, a digit map must not include a named embedded digit map that references itself.

To bar users from calling numbers that match a rule, add a `'!'` in front of that rule in the digit map. The rule is then referred to as a barring rule.

Examples:

- `1408xxxxxxx` – Matches any 11-digit number that starts with 1408.
- `011xx.` – Matches any number that starts with 011 followed by one or more digits.
- `<1408>xxxxxxx` – Matches any 7-digit number. The device prepends 1408 to the number when making the call.
- `<:1408>xxxxxxx` – Equivalent to the last example.
- `<+>1xxxxxxxxxx` – Prepends '+' to any 11-digit number that starts with 1.
- `<1:1408>xxxxxxx` – Matches any number that starts with 11408 followed by 7 digits. The device removes the 11 prefix when making the call.
- `*74(x|xx)` – Matches any number that starts with *74, followed by 1 or 2 digits.
- `**1(Msp1)` – Matches any number that starts with **1 and with the rest of digits matching the **DigitMap** in the SP1 Service.
- `<:1234>` – Matches an empty phone number and replaces with 1234. This is the syntax for a hotline to 1234.
- `<S0:1234>` – Equivalent to the last example.
- `<:1234>` – Hotline to the number 1234.
- `<S0:1234>` – Equivalent to the last example.
- `<S4:1234>` – Call 1234 if no digits entered for 4 seconds. This is the syntax of a warm line.
- `xx.853 7683` – Matches any number with at least 8 digits and ends with 8537683, such as 15108537683, 98537683.
- `(x.408 223 1122)` – Matches any number with at least 10 digits and ends with 408 223 1122, such as 4082231122 or 1408 223 1122.
- `xx.<#>` – Adds a # to the end of any number with 1 or more digits.
- `!1900xxx xxxx` – Barring all 11-digit numbers that start with 1900.
- `[^*]@.` – Arbitrarily long alphanumeric sequence (except #) that doesn't start with *
- `xx?` – Any 1- or 2-digit number.
- `(1xxxxxxxxxxxS0|xx.)` – Arbitrarily long digit sequence not starting with 1. Otherwise it is limited to 11 digits.

Matching Against Multiple Rules in a Digit Map

One important function of a digit map is to determine if you dialed sufficient digits during dialing. A digit map normally contains more than one rule. The Digit Map Processor (DMP) must return the best matched rule at some point, or declare that the input digit sequence is invalid. The DMP keeps refining its decision as

each digit is entered until it reaches a final decision, or is forced to make a timely decision when the interdigit timer expires.

The DMP restarts the interdigit timer on every newly entered digit. The duration of this timer can be either long or short. The long and the short timer values are set by default to 10 seconds and 2 seconds, respectively, and are configurable per Phone port via the **DigitMapLongTimer** and **DigitMapShortTimer** parameters. Whether to use the long or short interdigit timer depends on the current rule matching states. The DMP maintains a matching state for each rule in the digit map as it processes each input digit. The following states are defined:

- **Partially Matched (PM)** – The rule partially matches the accumulated input sequence. Initially all rules are in this state before any digit is entered. Rules in this state have the potential of becoming EM or IM as more digits are entered. Example: 1234 partially matches the rules `xxxxxxx`, `1xxxx`, `1234567`, `<123:>xxxx`.
- **Exactly Matched (EM)** – The rule exactly matches the accumulated input sequence. However, any further input digit turns this rule into the MM state. Example: 1234 exactly matches the rules `xxxx`, `1234`, `1xxx`, `<123:5678>x`.
- **Indefinitely Matched (IM)** – The rule matches the accumulated input sequence indefinitely, with a variable length such that the rule can potentially stay as IM as more matching digits are entered. Example: 011853 indefinitely matches the rules `xx.`, `011xx.`, `<011:>xx`.
- **Mismatch (MM)** – The rule doesn't match the accumulated input sequence. This state won't change as more digits are entered. Example: 1234 mismatches the rules `123`, `1xx`, `12345`.

Rules in the EM or IM state are candidates to be selected by the DMP. After processing a new digit, the DMP returns a final decision if any of the following conditions holds:

- All rules are the MM state. The DMP returns an error.
- One or more rules are in the EM state with no rules in the IM state. DMP returns the best matched EM rule. If the best matched rule is a barring rule, DMP returns an error instead.

Otherwise, DMP starts the short interdigit timer if there is at least one rule in the EM state, or else the long one. When the interdigit timer expires, DMP makes a timely decision by returning the best matched rule at that moment if one is found, or else a timeout error. Again if the best matched rule in this case is a barring rule, DMP returns an error instead. Note that the timer to wait for the first input digit isn't governed by the interdigit timer, but the duration of dial tone being played and could be a lot lengthier than the long interdigit timer.

The best matched rule is the one that has the most specific literals matching the input digit sequence. For example, the input sequence 1234 matches the rule `123x` better than `1xxx`. On the other hand, an EM rule is always selected over an IM rule.

Finally, the default interdigit timer can be overridden by appending the `Sn` element at the end of the rule ($n = 0-9$).

Consider this simple digit map:

```
(<1408>xxx xxxx)
```

As soon as 7 digits have been entered, the DMP returns a complete number by prepending the accumulated digits with 1408.

Consider another simple map:

```
(xx.)
```

After you dial one or more digits, the DMP returns the accumulated digits as a complete number when the long interdigit timer expires.

Combine the last two maps:

```
(xx. | <1408>xxx xxxx)
```

After you dial one or more digits (but fewer than seven digits), the DMP returns the accumulated digits as a complete number when the (long) interdigit timer expires. As soon as seven digits are entered, the DMP returns 1408 followed by the accumulated seven digits when the (short) interdigit expires. On the eighth digit and beyond, however, the DMP considers the first rule only and returns the accumulated digits as-is when the (long) interdigit timer expires.

Now add an S4 timer to the second rule:

```
(xx. | <1408>xxx xxxxs4)
```

In this case, the DMP behaves exactly the same as the last, except that the short interdigit timer the DMP uses upon receiving the seventh digit is overridden by a 4-second timer. Thus you have as long as 4 seconds instead of 2 to dial the eighth digit.

Force an Interdigit Timeout With a Pound(#) Key

When dialing, you can force an interdigit timeout with a # key instead of waiting for the DMP to timeout its own long or short timer. This is allowed as long as the # key doesn't match the current element of any PM rules. Otherwise the # key is consumed by the DMP instead of triggering a timeout.

Consider the digit map (33xx.)

If you enter 333#, the DMP immediately returns the number 333.

Now consider the digit map (33xx.|333#1234x.)

If you enter 333#, the DMP won't return, but continues to wait for further input or for its interdigit timer to expire. Note that the first rule "33xx." is now in the MM state since the digit # doesn't match "x". You can continue to enter 1234#, or 1234 and wait for a long interdigit timeout for the DMP to successfully return 333#1234.

Invoke a Second Dial Tone in a Digit Map

You can tell the device to start a tone after a certain pattern of digits have been dialed by specifying the element {t=<tone>} within a digit map, where <tone> is a 1- to 3-letter name of the tone to play. The tone stops when the next digit is entered. For example:

```
(**1{t=di2}(Msp)|**8{t=od}(Mli))
```

tells the device to play Second Dial Tone when **1 is dialed, or play Outside Dial Tone when **8 is dialed. Here is a full list of acceptable (case-insensitive) values of <tone>:

- bu = Busy Tone
- cf = Call Forwarded Dial Tone
- cm = Confirmation Tone
- co = Conference Tone
- cw1 - cw10 = Call Waiting Tone 1-10
- di = Dial Tone
- di2 = Second Dial Tone
- fb = Fast Busy Tone
- ho = Holding Tone

- ## Change an Interdigit Long Timer Dynamically After a Partial Match

```
(011 853 xxxx xxxx{L=5}x. |xx.)
```

User-Defined Digit Maps

- **Label:** An arbitrary string for referencing this digit map in other digit map specification. The value should be 2 to 16 characters long. For example, “friends”. In this case, (Mfriends) can be referenced in other digit maps, such as **PHONE Port::DigitMap**.
- **DigitMap:** A digit map to restrict the numbers that can be dialed or called with this service. Maximum length is 511 characters.

A User-Defined Digit Map For IPv4 Dialing

- Label: ipd
- Digit Map:
(xx.<*: @>xx?x?<*: .>xx?x?<*: .>xx?x?<*: .>xx?x? | xx.<*: @>xx?x?<*: .>xx?x?<*: .>
xx?x?<*: .>xx?x?<*: :>xx?x?x?x?)

- `<user-id>*<a>*<b>*<c>*<d>`
- `<user-id>*<a>*<b>*<c>*<d>*<port>`

where `<user-id>` is an arbitrary length numeric user-id, such as 100345, `<port>` is a port number in the range 0–65535, and each of `<a>`, `<b>`, `<c>`, `<d>` is a 1- to 3-digit pattern in the range 1–255 that identifies one byte of an IP address. The dialed number is translated into `<user-id>@<a>.<b>.<c>.<d>` and `<user-id>@<a>.<b>.<c>.<d>:<port>`. Here are some examples:

1234*192*168*15*113 maps to 1234@192.168.15.113

123456*192*168*15*180*5061 maps to 123456@192.168.15.180:5061

OBI Call Routing and Digit Map

Call routing is the process by which the device sets up a call bridge or a (endpoint) call based on such information as the trunk on which the call originates, the caller's number, the called number, and so forth. Call routing rules are parameters used to instruct the device how to route calls. A call can transform into a call bridge or an endpoint call after being routed by the device according to the given routing rules. Every call has to be originated from somewhere. From the device's perspective, calls originated from the trunk side are considered inbound calls, while calls originated from an endpoint are outbound calls. The call routing rule syntaxes for inbound calls and outbound calls are slightly different, and are explained in the following section.

A digit map serves to transform and restrict the number that can be dialed or called, and determine if you dialed sufficient digits to form a complete number. Each map is composed of one or more rules surrounded by parentheses (which MUST NOT be omitted).

Trunks, Endpoints, and Terminals

An OBI device is a Voice Service Bridge (VSB) that supports multiple voice services. It can bridge calls across any of the supported services. A call bridge refers to a voice connection connecting two calls on the same or different voice services. The device allows four concurrent independent call bridges. The following matrix shows the possible call bridge connections on the device.

Supported Two-Way Call Bridges

Supported Two-Way Call Bridges on the Device

	SP1	SP2	SP3	SP4	SP5	SP6	SP7	SP8	OBI TALK	OBI Blue Tooth 1	OBI Blue Tooth 2
SP1	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
SP2	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
SP3	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
SP4	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
SP5	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
SP6	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
SP7	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Supported Two-Way Call Bridges on the Device

	SP1	SP2	SP3	SP4	SP5	SP6	SP7	SP8	OBi TALK	OBi Blue Tooth 1	OBi Blue Tooth 2
SP8	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
SP9	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
OBi TALK	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
OBi Blue Tooth 1	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
OBi Blue Tooth 2	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No

Each supported service is also referred to as a trunk (a traditional telco term for a physical wire or wires that deliver phone services to homes or businesses). Each trunk is represented with two-letter abbreviation and a numeral-based instance identifier:

- SP1 = the SP1 voice service (with ITSP A, B, C, or D)
- SP2 = the SP2 voice service (with ITSP A, B, C, or D)
- SP3 = the SP3 voice service (with ITSP A, B, C, or D)
- SP4 = the SP4 voice service (with ITSP A, B, C, or D)
- SP5 = the SP5 voice service (with ITSP A, B, C, or D)
- SP6 = the SP6 voice service (with ITSP A, B, C, or D)
- SP7 = the SP7 voice service (with ITSP A, B, C, or D)
- SP8 = the SP8 voice service (with ITSP A, B, C, or D)
- SP8 = the SP8 voice service (with ITSP A, B, C, or D)
- PP1 = the OBiTALK service
- BT1 = OBiBlueTooth 1 service (with OBiBT dongle connected to USB Port 1)
- BT2 = OBiBlueTooth 2 service (with OBiBT dongle connected to USB Port 2)

The instance identifier can be omitted when it equals 1. Thus, BT is equivalent to BT1, PP is equivalent to PP1, and so forth. These short-hand notations are used heavily in configuring the device, as found in call routes, call forward numbers, and speed dials parameters. Unless stated otherwise, the abbreviated trunk names are case-insensitive.

The Phone ports, AA, AO, AI, and PK are the entities in an OBi5 Series device where calls can terminate (that is, start or end there), as opposed to the trunks, which rely on the corresponding service providers to terminate the call. In this guide, the Phone ports, AA, AO, AI, and PK are endpoints. Like the trunks, each endpoint is represented by a 2-letter abbreviation and a numeral-based instance identifier:

- PH n = the PHONE n port ($n = 1, 2, \dots, 8$)
- AA = the Auto Attendant
- AI = Audio Line In Port
- AO = Audio Line Out Port
- PK = The Call Park

Unless stated otherwise, abbreviated endpoint names are case-insensitive. A trunk or an endpoint is also referred to as a terminal in this guide.

The following matrix shows the possible call connections between the endpoints and the trunks.

Supported Endpoint Calls on the Device

	Any Trunk	PH1	PH2	PH3	PH4	PH5	PH6	PH7	PH8	AA	AI callee	AO callee	PK callee
Any Trunk	N/A	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PH1	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PH2	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PH3	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PH4	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PH5	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PH6	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
PH7	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
PH8	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes
AA	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes
AI caller	No	No	No	No	No	No	No	No	No	No	No	No	No
AO caller	No	No	No	No	No	No	No	No	No	No	No	No	No
PK caller	No	No	No	No	No	No	No	No	No	No	No	No	No

Hunt Groups

A Hunt Group is a list of terminal-sets that are alerted for an incoming call. The terminal-sets are alerted sequentially, one at a time. A terminal-set can include one or more terminals that are alerted simultaneously when the terminal-set is alerted. A phone port PH n and SP n ($n = 1, 2, 3, \dots$), are examples of a terminal.

As many as 10 hunt groups can be defined in an OBi5xx device using the **Hunt Groups::Group n** (for $n = 1, 2, \dots, 10$) parameters. The device can be configured to route certain incoming calls to be handled by one of the hunt groups, using the inbound call route parameters, in a way that is no different from routing incoming calls to ring any one of the phone ports.

To hunt sequentially for an answer, each terminal-set should be set to ring for a finite duration, say x seconds, such that if none of the terminals answer the call within x seconds, the device moves on to ring the next terminal-set, or stops the hunt if there are no more terminal-sets to try. In the case when all the terminals in a set are busy, the device skips over the set and moves on immediately. On the other hand, if the intention of the hunt is to find the first terminal-set that isn't busy, it is perfectly acceptable to specify in the hunt group configuration to ring each terminal-set indefinitely.

The general format of a hunt group parameter value is (all case-insensitive):

```
GP(name)={ringlist};group-par;group-par;...
```

where

- **name** = arbitrary string with no spaces for referencing this hunt group. It can be a number such as 1 or a generic string such as sales.
- **ringlist** is made up of one or more terminal-set separated by a +, with the following general format:
 - `ringlist = terminal-set +terminal-set +....`
 - The terminal-sets are alerted sequentially, one at a time.
- **group-par** is a parameter to control how to alert the members in the ringlist. It has the format:
 - `group-parameter-name=value`
 - The following group-parameter-names and corresponding values are defined:
 - ◆ **du**: value is the (default) duration in number of seconds to alert the terminal-set in each pass, or * to alert each set indefinitely
 - ◆ **na**: value is the terminal or hunt group to alert if the call isn't answered by any terminal in the hunt group; this mechanism may be used to daisy-chain hunt groups
 - ◆ **np**: value is the number of passes to cycle through the ringlist. Default is 1.
- A **terminal-set** is made up of a comma-separated list of terminals that are alerted simultaneously when the terminal-set is alerted. It has the following general format:


```
terminal-set = terminal,terminal,;K;set-par;set-par;... where
```

 - **set-par** is a parameter that controls the terminal-set. It has the format:


```
set-parameter-name=value
```

The following set-parameter-names and corresponding values are defined:

 - ◆ **du**: value is the duration in number of seconds to alert the terminal-set, or * to alert the set indefinitely. If this parameter isn't specified for the set, it takes the value from the group's du parameter, if one is specified. Otherwise the set rings indefinitely

Tips: You can sprinkle the hunt group parameter value with spaces to make it more readable. Beware that all white spaces are removed prior to processing these parameter values (including the ones in the hunt group name).

Hunt Group Examples

- Ring ph1,...,ph8 simultaneously:


```
GP(support)={ph1,ph2,ph3,ph4,ph7,ph8}
```
- Ring ph1, p2, ph3 sequentially; the first phone port that rings grabs the call. If all phones are busy, ring 14089991234 (via SP1 Service):


```
GP(1)={ph1+ph2+ph3};na=sp1(14089991234)
```
- Ring each of ph1, ph2, and ph3 sequentially for 20 seconds, twice, then forwards the call to 14089991123 if no one answers:


```
GP(2)={ph1+ph2+ph3};dur=20;np=0;na=sp1(14089991123)
```


- Ring ph1, p2, and ph3 simultaneously for 30 seconds, then ring ph4 and ph5 simultaneously for 30 seconds, then forward the call to the hunt group named `general` if no one answers:

```
GP(sales)={ph1,ph2,ph3+ph4,ph5;du=30+ph6};du=20;na=gp(general)
```

- Same as the last example, but also override the `na` and `du` group parameters of the hunt group named `general`:

```
GP(sales)={ph1,ph2,ph3+ph4,ph5;du=30+ph6};du=20;na=gp(general;du=10;na=sp1(14089991234))
```

Routing Incoming Calls to Hunt Groups

By adding rules to an inbound call route parameter, you can send incoming calls to be handled by a hunt group. For example, add the rules: `{>18007771000:gp(sales)}, {gp(general)}` to the **SPn Service::X_InboundCallRoute** parameter, such that all calls to 18007771000 are handled by the hunt group named `sales`, and the other calls by the hunt group named `general`.

SIP Trunking

There are 9 ITSP Profiles and 9 SP Services available in each OBi device. Each SP Service can be configured as a SIP Trunk in the following sense:

- Single REGISTRATION per SP account
- Multiple DID numbers for incoming calls and outgoing calls
- Incoming call routing based on called DID numbers, to ring any combinations of phone ports simultaneously or a hunt group
- Static Call Forwarding according to DID numbers via hunt groups
- Outgoing caller identity based on the phone port that originates the call
- Option to insert `user=phone` for caller identity (for a DID number) in outgoing calls

SIP Trunk Configuration

Configuration of a SIP Trunk on SP1 (ITSP Profile A) on OBi device.

In this example, SP1 on ITSP Profile A is a SIP trunk.

Use Case:

- All 8 phone ports use SP1 for outgoing calls
- Each phone port has a DID number, +14089991001, +14089991002, ... +14089991008, respectively
- Let `mytrunkid` be the Pilot UserID of the SIP trunk

SIP Trunking Configuration

Parameter	Value	Description
ITSP Profile A– SIP::X_SpoofCallerID	true	Enable this option to let the device insert a UserID in the FROM header of outbound INVITE that is different from the SIP Account UserID
ITSP Profile A– SIP::X_UserEqPhone	true	Enable this option to let the device insert a ;user=phone (URI) parameter in the Request-URI and TO header of outbound INVITE (for all calls on SP1). Don't enable this option if the server doesn't expect to see this URI parameter.
SP1 Service::AuthUserName	mytrunkid	Configure the SIP Account credentials. If the UserID for authentication is different from the SIP UserID, the SIP UserID must be configured in the URI parameter and authentication UserID in the AuthUserName parameter.
SP1 Service::AuthPassword	The authentication password that corresponds to mytrunkid.	
SP1 Service::URI		
SP1 Service::X_InboundCall Route	{>(@.4089991001):ph1}, {>(@.4089991002):ph2},..., {>(@.4089991008):ph8}	Configure incoming calls to a specific DID number to ring the corresponding phone port only. The @. syntax matches any number of characters before the last 10 digits of the called number (for example: + or +1 or +001). Note that you can also ring any combination of the phone ports for a given called number, or ring a hunt group (GP) defined in the device configuration.
Phone 1 Port::OutboundCallRoute	{(Mpli):pli(+14089991001; context=abd.com;;user=phone>\$2)}	Calls made from each phone port to assume a different identify based on assigned DID number. This identify is shown as the UserID in the From header (and Remote-Party-ID header if enabled) of the outbound INVITE. The Contact header of the INVITE on the other hand uses the Pilot UserID (the SP Account USERID) as the UserID. \$2 represents the target number that you dial. Note that additional call routing rules may be added for other call destinations. Note that UserID parameters after the first ; in the UserID are included as is as part of the user ID, but parameters after the first ; ; are inserted as URI parameters.
Phone 2 Port::OutboundCallRoute	{(Mpli):pli(+14089991002>\$2)}	
...	{(Mpli):pli(+1408999100n>\$2)} where $n = 1, 2, \dots, 8$	
Phone 8 Port::OutboundCallRoute	{(Mpli):pli(+14089991008>\$2)}	
Phone 1 Port::PrimaryLine	SP1 Service	
Phone 2 Port::PrimaryLine	SP1 Service	All phone ports to use SP1 as the primary line (pli).
...	SP1 Service	
Phone 8 Port::PrimaryLine	SP1 Service	

Notes:

- You can tell the device to insert a user=phone URI parameter in the FROM header of outbound INVITE (and Remote-Party-ID header if enabled) when spoofing the calling UserID with a DID number, by including a ;;user=phone parameter after the spoofed userid in an OutboundCallRoute rule. In fact, everything in the rules argument after the first ;; is included as URI parameter(s).

Call Forward Settings Per Phone Port

The following table shows a set of call forward settings per phone port:

Call Forward Settings

Parameter	Description
Phone PortN – Call Forwarding::ForwardAll	Enables forwarding all calls unconditionally. Value must be true or false.
Phone PortN – Call Forwarding::ForwardAllNumber	The number to forward all calls to unconditionally. It may include the Caller-ID number to use as the UserID of the SIP/Diversion header. Examples: SP1 (1000) SP1 (+14089991003>1000) SP1 (+14089991003;;user=phone>1000) SP1 (+14089991003;context=test.com;;user=phone>1000)
Phone PortN – Call Forwarding::ForwardOnBusy	Enables forwarding calls when the phone port is busy. Value must be true or false.
Phone PortN – Call Forwarding::ForwardOnBusyNumber	The number to forward calls to when the phone port is busy. See ForwardAllNumber for examples.
Phone PortN – Call Forwarding::ForwardOnNoAnswer	Enables forwarding calls when the call isn't answered after a few rings. Value must be true or false.
Phone PortN – Call Forwarding::ForwardOnNoAnswerNumber	The number to forward calls to when the call isn't answered after a few rings. See ForwardAllNumber for examples.
Phone PortN – Call Forwarding::ForwardOnNoAnswerRingCount	The number of rings to wait before triggering call forwarding on no answer, where one ring is equivalent to six seconds

Note that call forwarding from a phone port is implemented only by bridging two calls together by the device. That is, the device maintains the call with the original incoming caller (in the ringing state), while making a new call to the call forward target number. When the target rings or answers (whichever happens first), it bridges the two call legs together. The SIP/INVITE to the call forward target will have the original caller's Caller ID in the FROM header, and the Request-UserID of the original SIP/INVITE in a DIVERSION header. If the original Request-UserID has a user=phone parameter, the DIVERSION header carries the same URI parameter as well.

DND Setting Per Phone Port

The following table shows a Do Not Disturb (DND) setting per phone port:

Call Forward Settings

Parameter	Description
Phone PortN – Calling Features::DoNotDisturbEnable	Enables Do Not Disturb on a phone port. Value must be true or false.

Using Star Codes to Change Per-Phone-Port Call Forward and DND Settings

Assuming a different Star Code Profile is used for each Phone port, you can define local star codes to enable/disable the call forward settings per phone port, as shown below:

For Phone Port n ($n = 1, 2, 3, \dots, 8$) using Star Code Profile x ($x = A, B, \dots, H$)

```
*72,Cfwd All, coll(PHn($Cfan)), set(PHn($Cfa),1)
*73,Disable Cfwd All, set(PHn($Cfa), 0)
*60,Cfwd Busy, coll(PHn($Cfbn)), set(PHn($Cfb),1)
*61,Disable Cfwd Busy, set(PHn($Cfb), 0)
*62,Cfwd No Ans, coll(PHn($Cfnn)), set(PHn($Cfn),1)
*63,Disable Cfwd No Ans, set(PHn($Cfn),0)
*78,Enable DND, set(PHn($Dnd),1)
*79,Disable DND, set(PHn($Dnd),0)
```

It is also possible to use the same Star Code profile to change the values of call forwarding and DND settings for each Phone port, with the generic Phone port versions of the corresponding Star Code variables. The Phone port whose settings are to be changed in this case is the one that the Star Code is dialed from:

```
*72,Cfwd All, coll($PhCfan), set($PhCfa,1)
*73,Disable Cfwd All, set($PhCfa, 0)
*60,Cfwd Busy, coll($PhCfbn), set($PhCfb,1)
*61,Disable Cfwd Busy, set($PhCfb, 0)
*62,Cfwd No Ans, coll($PhCfnn), set($PhCfn,1)
*63,Disable Cfwd No Ans, set($PhCfn,0)
*78,Enable DND, set($PhDnd,1)
*79,Disable DND, set($PhDnd,0)
```

MWI Notification Routing

Your OBi device can handle SIP/NOTIFY for MWI status based on the DID number in the Request-UserID of the NOTIFY request. The same processing is used inside or outside the context of a Subscription dialog. You can define routing rules to process the MWI status by any combination of phone ports. There are two settings for this under each **SPn Service–Calling Features::X_MWIRoute** parameter for basic MWI

handling (by playing Stutter Tone) and **SPn Service–Calling Features::X_VMWIRoute** for VMWI. Here is an example of a MWI Route:

```
{ (@.4089991001)>ph1}, { (@.4089991002)>ph2}, ..., { (@.4089991008)>ph8}, {ph7,ph8}
```

In the last example, SIP/NOTIFY sent for a particular DID number is handled by a corresponding phone port only, except the last rule which is a catch-all rule for SIP/NOTIFY sent for anything else.

Virtual Stacking

Virtual stacking is only applicable in OBi504/508 devices. There are two specific cases of virtual stacking:

- 1 All devices within the same subnet with static/reserved IP addresses.
- 2 All devices are scattered on the Internet and linked via OBiTALK.

In general, the two cases can be mixed where some devices are on the same subnet as the MU that also serves as their router (gateway), and other devices scattered elsewhere on the Internet. The architecture thus allows for example the MU and some EUs in the main office, but have one or more devices in branch offices.

Virtual Stacking is a set of configuration parameters that links a few OBi500 devices to provide more phone ports in the context of Hunt Groups and SIP Trunking. The linking of the devices can be achieved on the same LAN or across the Internet. The device supports virtually stacking as many as four devices to provide 32 phone ports. One of the devices must be designated as the Master Unit (MU), and the other devices are called Expansion Units (EU): EU1 for the first EU, EU2 for the second, and EU3 for the third.

Note that any OBi500 device can be MU or EU. Given four OBi500 devices to form a virtual stack, we recommend using the convention to assign devices as MU and EU based on their OBi numbers, such that the device with the smallest OBi number is the MU, the next smallest EU1, the next smallest EU2, and so on.

There are two applications that virtual stacking is particularly useful:

- Forming hunt groups that span as many as 32 phone ports
- SIP trunking with as many as 32 DID numbers to support a corresponding number of analog phone lines

We will discuss the device configuration for these applications in the following sections. With no loss of generality, let's assume the MU has a SIP trunk installed on SP1. On the MU, you must enable the Boolean parameter **Hunt Groups::VirtualStacking**.

Case 1: All Units Having Static or Reserved IP Addresses On the Same Subnet

Let the MU, EU1, EU2, and EU3 have static or reserved IP address `ipaddr0`, `ipaddr1`, `ipaddr2`, and `ipaddr3` respectively, on the same subnet. On the MU, pick any unused SPn Service ($n = 2, 3, \dots, 9$) to communicate with the EU. On the EU, pick any unused SPn Service ($n = 1, 2, 3, \dots, K$) to communicate with the MU. We use SP2 on the MU and SP1 on the EU in the current example.

Extending Hunt Groups on the MU to span 32 Phone Ports

On the MU, use the following terminals in the configuration of hunt groups:

- SP2(\$1>n@ipaddr1), $n = 1009-1016$ to hunt, respectively, PH1-PH8 ports on EU1.
- SP2(\$1>n@ipaddr2), $n = 1017-1024$ to hunt, respectively, PH1-PH8 ports on EU2.

- SP2(\$1>n@ipaddr3), n = 1025-1032 to hunt, respectively, PH1-PH8 ports on EU3.

On EU1, set **SP1 Service::X_InboundCallRoute** = {>1009:ph1}, {>1010:ph2}, ... {>1016:ph8}

On EU2, set **SP1 Service::X_InboundCallRoute** = {>1017:ph1}, {>1018:ph2}, ... {>1024:ph8}

On EU3, set **SP1 Service::X_InboundCallRoute** = {>1025:ph1}, {>1026:ph2}, ... {>1032:ph8}

Extending SIP Trunking on the MU to support 32 DID

- +14089991001 to +14089991008 are assigned to PH1–PH8, respectively, on the MU.
- +14089991009 to +14089991016 are assigned to PH1–PH8, respectively, on the EU1.
- +14089991017 to +14089991024 are assigned to PH1–PH8, respectively, on the EU2.
- +14089991001 to +14089991008 are assigned to PH1–PH8, respectively, on the EU3.

On EU1:

- Set **SP1 Service::X_InboundCallRoute** = {>1009:ph1}, {>1010:ph2}, ... {>1016:ph8}
- Set **Phone n Port::OutboundCallRoute** = {sp1 (m>\$2) } for n = 1–8 corresponding to m = 1009–1016, respectively

On EU2:

- Set **SP1 Service::X_InboundCallRoute** = {>1017:ph1}, {>1018:ph2}, ... {>1024:ph8}
- Set **Phone n Port::OutboundCallRoute** = {sp1 (m>\$2) } for n = 1–8 corresponding to m = 1017–1024, respectively

On EU3:

- Set **SP1 Service::X_InboundCallRoute** = {>1025:ph1}, {>1026:ph2}, ... {>1032:ph8}
- Set **Phone n Port::OutboundCallRoute** = {sp1 (m>\$2) } for n = 1–8 corresponding to m = 1025–1032, respectively

On the MU:

- Set **ITSP Profile A–SIP::X_SpoofCallerID** = true

- Set **SP1 Service::X_InboundCallRoute** =


```
{> (@.4089991001) :ph1},
{> (@.4089991002) :ph2},
...,
{> (@.4089991008) :ph8},
{> (@.4089991009) :sp2 ($1>1009@ipaddr1) },
{> (@.4089991010) :sp2 ($1>1010@ipaddr1) },
...,
{> (@.4089991016) :sp2 ($1>1016@ipaddr1) },
{> (@.4089991017) :sp2 ($1>1017@ipaddr2) },
{> (@.4089991018) :sp2 ($1>1018@ipaddr2) },
...,
{> (@.4089991024) :sp2 ($1>1024@ipaddr2) },
{> (@.4089991025) :sp2 ($1>1025@ipaddr3) },
{> (@.4089991026) :sp2 ($1>1026@ipaddr3) },
...,
{> (@.4089991032) :sp2 ($1>1032@ipaddr3) }
```
- Set **Phone n Port::OutboundCallRoute** =


```
{ (Msp1) :sp1 (+1408999100n;context=xyz.com;;user=phone>$2) }
for n = 1, 2, ..., 8
```
- Set **SP2 Service::X_InboundCallRoute** =


```
{1009:sp1 (+14089991009;context=xyz.com;;user=phone>$2) },
{1010:sp1 (+14089991010;context=xyz.com;;user=phone>$2) },
...,
{1032:sp1 (+14089991032;context=xyz.com;;user=phone>$2) },
```

On EU1, EU2, and EU3:

- Set **Phone n Port::PrimaryLine** = SP1 Service for all phone ports
- Set **ITSP Profile A::ProxyServer** = ipaddr0
- Set **ITSP Profile A::ProxyServerPort** = 5061 (Note: 5061 is the default **SP2 Service::X_UserAgent** Port)
- Set **ITSP Profile A-SIP::X_SpoofCallerID** = true
- Set **SP1 Service::RegisterEnable** = false

Extending SIP Trunking on the MU to support 32 Voicemail Accounts

Continuing from the last example, suppose further that the SIP Trunk has 32 voicemail accounts corresponding to the 32 DID numbers, with the voicemail account ID `mboxn`, for $n = 1, 2, \dots, 32$. When the voicemail server sends an MWI (a SIP/NOTIFY with Event:message-summary) to the device, it may indicate the mailbox account ID as the User ID field in the request URI of the SIP/NOTIFY, or in the Message-Account field in the message's body. The device looks for the ID in the Message-Account field first.

If this isn't found, it takes the ID from the request URI. The device routes the MWI message according to the X_MWIRoute parameter for the current SIP trunk (SP1 Service in this example), which may be extended to include routing the message to EUs. The following example routes MWI messages based on one-to-one mapping of the voicemail account to the Phone port.

On the MU:

- Set **SP1 Service::X_MWIRoute** =

```
{mbox1:ph1},{mbox2:ph2},;K,{mbox8:ph8},
{mbox9:sp2(1009@ipaddr1)},{mbox10:sp2(1010@ipaddr1)},...,{mbox16:sp2(1016@ipaddr1)},
{mbox17:sp2(1017@ipaddr2)},{mbox18:sp2(1018@ipaddr2)},...,{mbox24:sp2(1024@ipaddr2)},
{mbox25:sp2(1025@ipaddr3)},{mbox26:sp2(1026@ipaddr3)},...,{mbox32:sp2(1032@ipaddr3)}
```
- Set **SP1 Service::X_VMWIRoute** =

```
{mbox1:ph1},{mbox2:ph2},...,{mbox8:ph8},
```

ON EU1:

- Set **SP1 Service::X_MWIRoute** = {1009:ph1},{1010:ph2},...,{1016:ph8}
- Set **SP1 Service::X_VMWIRoute** = {1009:ph1},{1010:ph2},...,{1016:ph8}

ON EU2:

- Set **SP1 Service::X_MWIRoute** = {1017:ph1},{1018:ph2},...,{1024:ph8}
- Set **SP1 Service::X_VMWIRoute** = {1017:ph1},{1018:ph2},...,{1024:ph8 }

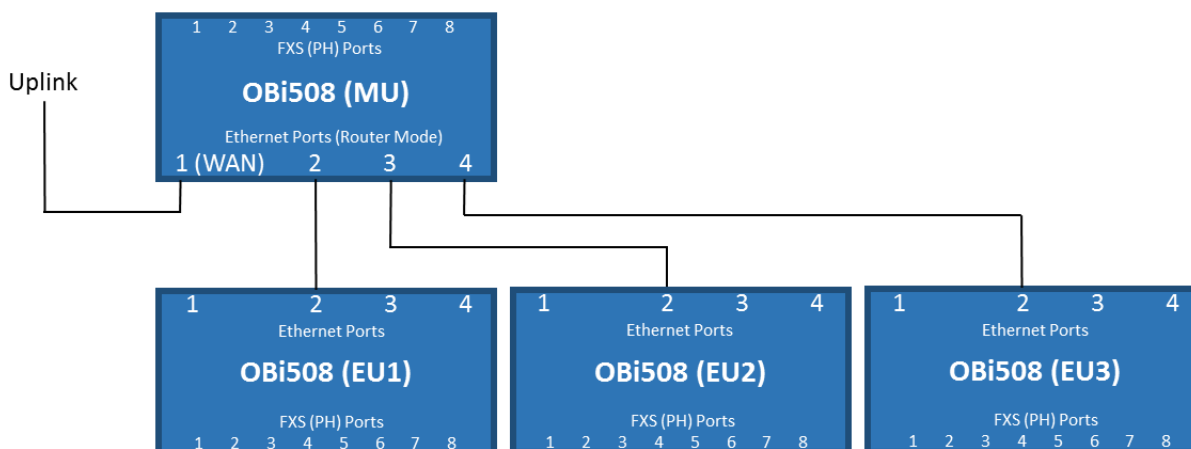
ON EU3:

- Set **SP1 Service::X_MWIRoute** = {1025:ph1},{1026:ph2},...,{1032:ph8}
- Set **SP1 Service::X_VMWIRoute** = {1025:ph1},{1026:ph2},;K...,{1032:ph8}

A More Specific Example

Given four OBi508 devices where the MU is configured as a router, and the three EUs are connected to the LAN ports of the MU as shown in the following figure.

Locally connected SIP trunking

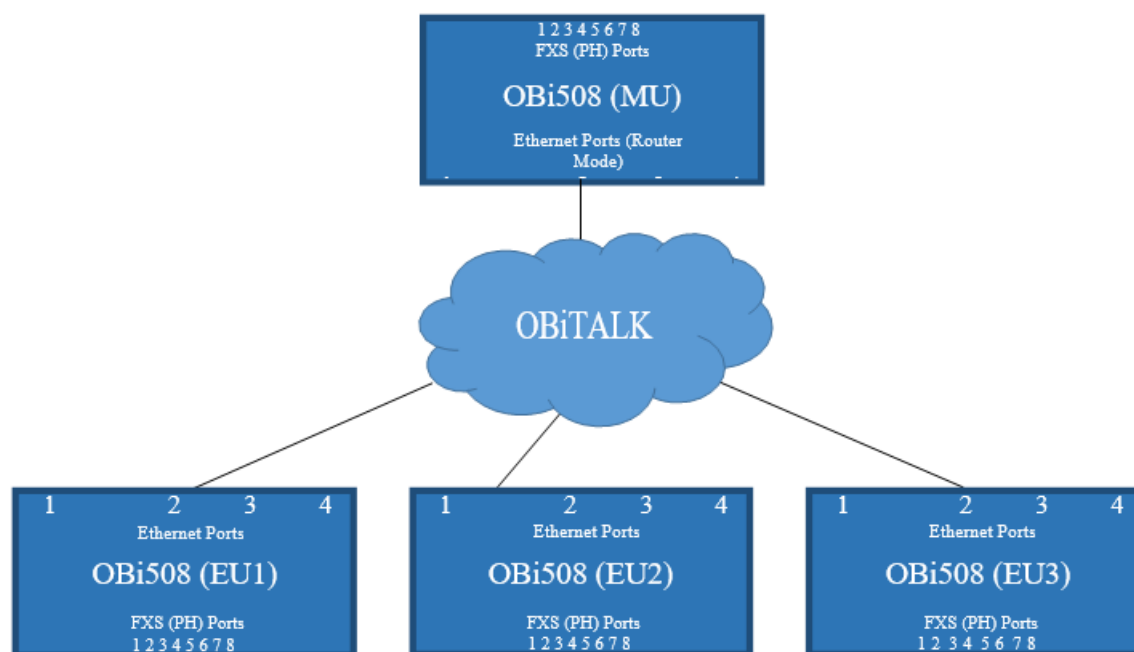


All EU devices have DHCP enabled and therefore will have IP addresses assigned by the DHCP server in the MU. Let `mac1`, `mac2`, and `mac3` be the MAC addresses of EU1, EU2, and EU3, respectively. We can reserve the IP addresses to be assigned to these 3 MACs on the MU. Hence, in referencing to the last example, `ipaddr0` would be the same as the value of **LAN Settings::RouterIPAddress** on the MU, and `ipaddr1`, `ipaddr2`, and `ipaddr3` would be equal to the addresses reserved for `mac1`, `mac2`, and `mac3`, respectively, on the MU.

Case 2: MU/EU Located Anywhere On the Internet

Let the MU, EU1, EU2, and EU3 have OBi Numbers `obin0`, `obin1`, `obn2`, and `obn3`, respectively, and reachable to PDMS-SP via the Internet.

Remotely connected SIP trunking



Extending Hunt Groups on the MU to Span 32 Phone Ports

On the MU, use the following terminals in the configuration of hunt groups:

- VG1(*n*), *n* = 1009–1016 to hunt, respectively, PH1–PH8 ports on EU1
- VG2(*n*), *n* = 1017–1024 to hunt, respectively, PH1–PH8 ports on EU2
- VG3(*n*), *n* = 1025–1032 to hunt, respectively, PH1–PH8 ports on EU3
- **Voice Gateway1::AccessNumber** = obn1
- **Voice Gateway2::AccessNumber** = obn2
- **Voice Gateway3::AccessNumber** = obn3
- **Voice Gateway1::SpoofCallerID** = true
- **Voice Gateway2::SpoofCallerID** = true
- **Voice Gateway3::SpoofCallerID** = true

On EU1, set **OBiTALK Service::X_InboundCallRoute** = {>1009:ph1}, {>1010:ph2}, ... {>1016:ph8}

On EU2, set **OBiTALK Service::X_InboundCallRoute** = {>1017:ph1}, {>1018:ph2}, ... {>1024:ph8}

On EU3, set **OBiTALK Service::X_InboundCallRoute** = {>1025:ph1}, {>1026:ph2}, ... {>1032:ph8}

Extending SIP Trunking on the MU to support 32 DID Numbers

Let the DID numbers be: +14089991001, +14089991002, ... +14089991032, such that:

- +14089991001 ... +14089991008 are assigned to PH1–PH8, respectively, on the MU
- +14089991009 ... +14089991016 are assigned to PH1–PH8, respectively, on the EU1
- +14089991017 ... +14089991024 are assigned to PH1–PH8, respectively, on the EU2
- +14089991001 ... +14089991008 are assigned to PH1–PH8, respectively, on the EU3.

On EU1:

- Set **OBiTALK Service::X_InboundCallRoute** = {obn0>1009:ph1}, {obn0>1010:ph2}, ... {obn0>1016:ph8}
- Set **Phone *n* Port::OutboundCallRoute** = {vg1 (*m*>\$2) } for *n* = 1–8 corresponding to *m* = 1009–1016, respectively.

Alternatively, using PP directly without going VG, set **Phone *n* Port::OutboundCallRoute** = {PP (*m*@obn0>\$2) } for *n* = 1–8 corresponding to *m* = 1009–1016

On EU2:

- Set **OBiTALK Service::X_InboundCallRoute** = {obn0>1017:ph1}, {obn0>1018:ph2}, ... {obn0>1024:ph8}
- Set **Phone *n* Port::OutboundCallRoute** = {vg1 (*m*>\$2) } for *n* = 1–8 corresponding to *m* = 1017–1024, respectively.

Alternatively, using PP directly without going VG, set **Phone *n* Port::OutboundCallRoute** = {PP (*m*@obn0>\$2) } for *n* = 1–8 corresponding to *m* = 1017–1024

On EU3:

- Set **OBiTALK Service::X_InboundCallRoute** = {obn0>1025:ph1},{obn0>1026:ph2},...
{obn0>1032:ph8}
- Set **Phone n Port::OutboundCallRoute** = {vg1(m>\$2)} for n = 1–8 corresponding to m = 1025–1032, respectively.

Alternatively, using PP directly without going VG, set **Phone n Port::OutboundCallRoute** = {PP(m@obn0>\$2)} for n = 1–8 corresponding to m = 1025–1032

On EU1, EU2, and EU3:

- Set **Voice Gateway 1::AccessNumber** = obn0
- Set **Phone n Port::PrimaryLine** = Trunk Group 1 for all phone ports
- Set **Trunk Group 1::TrunkList** = vg1
- Set **Voice Gateway 1::SpoofCallerID** = true

On the MU:

- Set **SP1 Service::X_InboundCallRoute** =

```
{>(@.4089991001):ph1},
{>(@.4089991002):ph2},
...,
{>(@.4089991008):ph8},
{>(@.4089991009):VG1($1>1009)},
{>(@.4089991010):VG1($1>1010)},
...,
{>(@.4089991016):VG1($1>1016)},
{>(@.4089991017):VG2($1>1017)},
{>(@.4089991018):VG2($1>1018)},
...,
{>(@.4089991024):VG2($1>1024)},
{>(@.4089991025):VG3($1>1025)},
{>(@.4089991026):VG3($1>1026)},
...,
{>(@.4089991032):VG3($1>1032)}
```

Alternatively, you may use PP directly without going through VG, albeit somewhat lengthier:

SP1 Service::X_InboundCallRoute =

```
{> (@.4089991001) :ph1},
{> (@.4089991002) :ph2},
...,
{> (@.4089991008) :ph8},
{> (@.4089991009) :PP ($1>1009@obn1) },
{> (@.4089991010) :PP ($1>1010@obn1) },
...,
{> (@.4089991016) :PP ($1>1016@obn1) },
{> (@.4089991017) :PP ($1>1017@obn2) },
{> (@.4089991018) :PP ($1>1018@obn2) },
...,
{> (@.4089991024) :PP ($1>1024@obn2) },
{> (@.4089991025) :PP ($1>1025@obn3) },
{> (@.4089991026) :PP ($1>1026@obn3) },
...,
{> (@.4089991032) :PP ($1>1032@obn3) }
```

- Set **Phone n Port::OutboundCallRoute =**

{ (Msp1) :sp1 (+1408999m;context=xyz.com;;user=phone>\$2) } for n = 1, 2, ..., 8
corresponding to m = 1001–1009, respectively

- Set **OBiTALK Service::X_InboundCallRoute =**

```
{1009@obn1:sp1 (+14089991009;context=xyz.com;;user=phone>$2),
{1010@obn1:sp1 (+14089991010;context=xyz.com;;user=phone>$2),
...,
{1016@obn1:sp1 (+14089991010;context=xyz.com;;user=phone>$2),
{1017@obn2:sp1 (+14089991017;context=xyz.com;;user=phone>$2),
...,
{1024@obn2:sp1 (+14089991024;context=xyz.com;;user=phone>$2),
{1025@obn3:sp1 (+14089991025;context=xyz.com;;user=phone>$2),
...,
{1032@obn3:sp1 (+14089991032;context=xyz.com;;user=phone>$2)}
```

Note that for OBiTALK service, caller-id matching is based on the caller's OBi number. To match the caller's number, use the syntax `number@OBiNumber` to match both the caller's number and OBi number, or `number@` to match just the caller's number, in the caller-id field inside PP() in the inbound call route. Note also that when routing a call to go out from PDMS-SP (PP), the caller number and caller OBi number are always spoofed; no setting is required. On the other hand, when routing call to go out via VG, the caller id isn't spoofed by default. You must enable the SpoofCallerID option on the VG to spoof the caller-id.

Extending SIP Trunking on the MU to Support 32 Voicemail Accounts

Continuing from the last example, suppose further that the SIP Trunk has 32 voicemail accounts corresponding to the 32 DID numbers, with the voicemail account ID mboxn, for n = 1, 2, ..., 32. When the voicemail server sends an MWI (a SIP/NOTIFY with Event:message-summary) to the device, it may indicate the mailbox account ID as the User ID field in the request URI of the SIP/NOTIFY, or in the Message-Account field in the message's body. The OBi5xx looks for the ID in the Message-Account field first. If this isn't found, it takes the ID from the request URI. The OBi5xx routes the MWI message according to the X_MWIRoute parameter for the current SIP trunk (SP1 Service in this example), which may be extended to include routing the message to EUs. Below is an example that routes MWI message based on one-to-one mapping of the voicemail account to the Phone port.

On the MU:

- Set **SP1 Service::X_MWIRoute** =

```
{mbox1:ph1},{mbox2:ph2},iK,{mbox8:ph8},
{mbox9:pp(1009@obn1)},{mbox10:pp(1010@obn1)},...,{mbox16:pp(1016@obn1)
},
{mbox17:pp(1017@obn2)},{mbox18:pp(1018@obn2)},...,{mbox24:pp(1024@obn2)
}),
{mbox25:pp(1025@obn3)},{mbox26:pp(1026@obn3)},...,{mbox32:pp(1032@obn3)
})
```
- Set **SP1 Service::X_VMWIRoute** =

```
{mbox1:ph1},{mbox2:ph2},iK,{mbox8:ph8},
```

ON EU1:

- Set **OBiTALK Service::X_MWIRoute** = {1009:ph1},{1010:ph2},iK,{1016:ph8}
- Set **OBiTALK Service::X_VMWIRoute** = {1009:ph1},{1010:ph2},iK,{1016:ph8}

ON EU2:

- Set **OBiTALK Service::X_MWIRoute** = {1017:ph1},{1018:ph2},iK,{1024:ph8}
- Set **OBiTALK Service::X_VMWIRoute** = {1017:ph1},{1018:ph2},iK,{1024:ph8}

ON EU3:

- Set **OBiTALK Service::X_MWIRoute** = {1025:ph1},{1026:ph2},iK,{1032:ph8}
- Set **OBiTALK Service::X_VMWIRoute** = {1025:ph1},{1026:ph2},iK,{1032:ph8}

Service Providers

Voice Services

- SP1–8
- OBiTALK
- AA
- Gateways and Trunk Groups
- OBiBluetooth 1–2
- Hunt Groups

SIP Service Provider Features

You can configure up to four SIP accounts or SIP Trunks on the device. For the purposes of this guide and elsewhere on the system web interface, documentation, and the PDMS-SP portal, the term ITSP describes the logical entity providing the SIP Trunk service to the device. When the device is used with an IP PBX, the IP PBX takes the place of the ITSP if it is the entity providing the SIP Trunk account credential and connectivity to the device.

Each ITSP configuration is grouped together as an ITSP Profile, referred to as ITSP Profiles A, B, C, and D. On the other hand, the SP service account specifics are grouped under the heading **SP n Service**, where $n = 1–8$. An ITSP Profile includes such parameters as **ProxyServer**, **OutboundProxy**, and **DigitMap**, but doesn't include account-specific parameters. An SP Service includes account-specific parameters such as **AuthUserName** (usually the phone number of the account), **AuthPassword**, **CallerIDName**, and **X_ServProfile** (which ITSP Profile to assume). If the SP Services use the same ITSP, then only one ITSP Profile needs to be configured with all SP Services referred to the same profile.

From the device point of view, the SP n Service using ITSP Profile X is enabled with the following minimal settings:

- **ITSP Profile X – SIP::ProxyServer** = *Not Blank*
- **SP n Service::Enabled** = Yes
- **SP n Service::AuthUsername** = *Not Blank*

where $X = A$ or B , $n = 1–8$. Otherwise, the service is considered disabled.

SIP Registration

Devices can be set periodically register with a SIP Proxy Server or SIP Registration Server. SIP Proxy Server and SIP Registration Server can be different, although they are usually the same in practice. SIP Proxy Server is a required parameter that must be configured on the device. The Registration Server is optional and assumed to be the same as the SIP Proxy Server if it isn't configured on the device.

The main purpose of registration is to create and maintain a dynamic binding of the SIP account to the device's local contact address. The service provider can also rely on this periodic message to infer if the device is online and functional. Each device takes only one local IP address that is either statically assigned in the device's configuration, or dynamically obtained from a local DHCP server. The SP n services (for $n = 1$ through 8) each use a different local contact port for sending and receiving SIP messages (defaults are 5060, 5061, 5062, and 5063).

Note that dynamic address binding through periodic registration isn't strictly necessary if the local IP address of the device doesn't change. The device's contact address can be statically configured on the Registration Server.

SIP Outbound Proxy Server

An outbound proxy server can be configured on the device such that all outbound requests are sent via the outbound proxy server instead of directly to the SIP Proxy Server or Registration Server.

If the outbound proxy server is listening at a non-standard port, the correct port value must be specified in the **OutboundProxyPort** parameter. The **OutboundProxy** can use a different transport protocol from the **ProxyServer**. The transport protocol to use to communicate with the **OutboundProxy** can be set in the **OutboundProxyTransport** parameters. If **OutboundProxyTransport** is TCP or TLS, your device initiates a TCP or TLS connection only with the **OutboundProxy**. All subsequent messages exchanged between your device and the servers MUST use the same connection. If for any reason the connection is closed, your device attempts to re-establish the connection with the **OutboundProxy** following an exponential back-off retry pattern.

Even though your device only exchanges messages directly with the **OutboundProxy**, the **ProxyServer**, **ProxyServerPort**, and **ProxyServerTransport** parameters are still very much relevant and important since the SIP requests sent by your handset to the server are formed based on these values, not based on the **OutboundProxy** value. The **OutboundProxy** value should never appear in the SIP requests generated by your device, unless the **OutboundProxy** parameter has the same value as **ProxyServer**.

Some server implementations include the outbound proxy server in a Record-Route header such that your device should not respect the locally configured **OutboundProxy** value after the initial INVITE is sent for a new call. This behavior can be achieved by enabling the **ITSP Profile X – SIP::X_BypassOutboundProxyInCall** option. However, this option has no effect when the **OutboundProxyTransport** is TCP or TLS, as your device always uses the same connection to send messages to the server.

DNS Lookup of SIP Servers

When sending out SIP requests to the server, the device looks up the IP address of the server using standard DNS query if the server is specified as a domain name instead of an IP address. If an Outbound Proxy Server is configured, it's used instead of the SIP Proxy Server or SIP Registration Server. The resolution of the server domain name into IP address is performed in the following manner:

- Try looking up the name as DNS A Record. If not found,
- Try looking up the name as DNS SRV Record. If not found,
- Try looking up the name as DNS SRV Record with “_sip._udp.” prepended to the host name. If not found, fail the request.

If the result from the DNS query is an SRV record, the server port is taken from that record also. The server port value configured on the device is ignored. Otherwise, the server port is taken from the configured value or uses port 5060 if none is specified.

NAT Traversal Considerations

If the device sits behind a NAT router (typically the case), it can discover the mapped external address corresponding to its local SIP contact address as seen by the server in one of the following ways:

- From the “received=” and “rport=” parameters of the VIA header of the REGISTER response sent by the server. These two parameters tell the device its mapped IP address and port number. This method is used if periodic registration is enabled on the device.
- From the response to a STUN binding request the device sent to a STUN server. This method is used by enabling **X_KeepAliveEnable** and setting **X_KeepAliveMsgType** to “stun”. The keep-alive messages are sent to the same server where a REGISTER request would be sent.

The device always uses the mapped external contact address in all outbound SIP requests instead of its local contact address if one is discovered by either method discovered above.

SIP Proxy Server Redundancy and DualREGISTRATION

Server Redundancy specifically refers to the device’s capability to:

- Look for a working server to REGISTER with from among a list of candidates.
- Switch to another server once the server that it currently registers with becomes unresponsive. In other words, device registration must be enabled to use the server redundancy feature.

Other SIP requests, such as INVITE or SUBSCRIBE, are sent to the same server that the device currently registers with.

If Outbound Proxy Server is provided, server redundancy is applied to the Outbound Proxy Server instead of the REGISTRATION server. Server redundancy behavior is enabled by enabling the **ITSP Profile X – SIP::X_ProxyServerRedundancy** parameter, which is disabled by default.

Another requirement for using the server redundancy feature is that the underlying server must be configured in the device as a domain name instead of an IP address. This allows the device to collect a list of candidate servers based on DNS query.

The domain name can be looked up as DNS A record or DNS SRV record. For A records, all the IP addresses returned by the DNS server are considered to have the same priority. For SRV records, the hosts returned by the DNS server can be each assigned a different priority.

After a list of candidate servers are obtained, the device first looks for a working server according to the stated priority. A *working server* means one that the device can successfully register with. This is known as the *Primary Server*. Subsequently, the device maintains registration with the primary server the usual way. However, if no working server is found after traversing the entire list, device takes a short break and repeats the search in the same order.

While maintaining registration with the Primary Server, the device continually attempts to fall back to one of the candidate servers that has higher priority than the primary server, if any. The list of candidate servers that the device is trying to fall back on is known as the *primary fallback list*, which may be empty.

In addition, the device can be configured to maintain a secondary registration with a server that has lower or equal priority than the primary server. Secondary registration can be enabled by setting the parameter **X_SecondaryRegistration** to YES. If **X_ProxyServerRedundancy** is NO, however,

X_SecondaryRegistration doesn't take any effect. If this feature is enabled, as soon as a primary server is found, the device searches for a working secondary server in the same manner from the list of candidate servers that are of lower or equal priority than the primary server. Similarly, once a secondary server is found, the device forms a *secondary fallback list* to continually attempt to fall back on if the list isn't empty.

The intervals for checking the primary fallback list and the secondary fallback list are configured in the **X_CheckPrimaryFallbackInterval** and **X_CheckSecondaryFallbackInterval** parameters. These parameters are specified in seconds and the default value is 60 for both.

Notes:

- Existence of a secondary server implies a primary server exists.
- If the secondary server exists, it immediately becomes the primary server when the current primary server fails. The device then starts searching for a new secondary server if the candidate set isn't empty.
- The candidate list can change (be lengthened, shortened, have its priority changed, and so forth) on every DNS renewal (based on the entry's TTL). The device rearranges the primary and secondary servers and fallback lists accordingly, whichever applies.

If the server redundancy feature is disabled, the device resolves only one IP address from the server's domain name, and won't try other IP addresses if the server isn't responding.

SIP Privacy

The device observes inbound caller privacy and decodes caller's name and number from SIP INVITE requests by checking the FROM, P-Asserted-Identity (PAID for short), and Remote-Party-ID (RPID for short) message headers. All these headers can carry caller's name and number information.

If PAID is present, device takes the name and number from it. Otherwise, it takes the name and number from RPID if it is present, or from the FROM header otherwise. RPID, if present, includes the privacy setting desired by the caller. This privacy can indicate one of the following options:

- *off* = no privacy requested. The device shows name and number.
- *full* = full privacy requested. The device hides both name and number.
- *name* = name privacy requested. The device shows the number but hides the name.
- *uri* = uri privacy requested. The device shows the name but hides the number.

Regardless, if PAID exists or not, the device always takes the privacy setting from the RPID if it's present in the INVITE request. Note that if the resulting caller name is "Anonymous" (case-insensitive), device treats it as if the caller is requesting full privacy.

For outbound calls, caller's preferred privacy setting can be stated by the device in a RPID header of the outbound INVITE request. To enable this behavior, the **ITSP Profile X – SIP::X_InsertRemotePartyID** parameter must be set to YES or TRUE, which is the default value of this parameter. The device supports only two outbound caller privacy settings: *privacy=off* or *privacy=full*. The RPID header generated by the device carries the same name and number as the FROM header. If outbound caller-ID is blocked, the device sets *privacy=full* in RPID, and also sets the display name in the FROM and RPID headers to "Anonymous" for backward compatibility. The device won't insert PAID in outbound INVITE requests.

STUN and ICE

The device supports standard STUN based on RFC3489 and RFC5389 for passing inbound RTP packets to the device sitting behind NATs. The parameters that control the STUN feature are found in the **ITSP**

Profile X – General:: section:

- **STUNEnable** – Enables this feature (default is NO or FALSE).
- **STUNServer** – The IP address or domain name of the external STUN server to use. STUN feature is disabled if this value is blank, which is the default.
- **X_STUNServerPort** – The STUN Server's listening UDP port. Default value is 3478 (standard STUN port).

The STUN feature used in this context is only for RTP packets, not SIP signaling packets, which typically don't require STUN. The device sends a STUN binding request right before making or answering a call on SP1/2. If the request is successful, the device decodes the mapped external address and port from the binding response and uses them in the m= and c= lines of its SDP offer or answer sent to the peer device. If the request fails, such as STUN server not found or not responding, the call goes on without using external address in the SDP.

Standard RTP requires the use of an even-numbered port in the m= line. If the external port isn't an even number, the device changes the local RTP port and redoes STUN, and continues to do this as many as four times or until an even external port number is found. If the fourth trial still results in an odd external port number, the call goes on without using an external address in the SDP.

The device supports standard ICE based on RFC5245. ICE is done on a per-call basis for automatically discovering which peer address is the best route for sending RTP packets. To enable ICE on the device, set the **ITSP Profile X – General::X_ICEEnable** parameter to YES (or TRUE). The default is NO (or FALSE).

ICE is effective if STUN is also enabled. However, STUN not a requirement for using ICE on the device. If STUN is enabled and an external RTP address different from its local address is discovered, the device offers two ICE candidates in its SDP:

- The local (host) address (highest priority)
- The external (srflx or server reflexive) address

Otherwise, only the local host candidate is shown in the device's SDP. Note that the device uses the srflx address in the m= and c= lines of the SDP if STUN is enabled and successful.

If ICE is enabled and the peer's SDP has more than one candidate, the device sends STUN requests to each peer candidate from its local RTP port. As soon as it receives a response from the highest priority candidate, the device concludes ICE and uses this candidate to communicate with the peer subsequently. Otherwise, the device allows as long as 5 seconds to wait for the response from all the candidates, and selects the highest priority candidate that has a response. Once ICE completes successfully, the device further applies symmetric RTP concept to determine the peer's RTP address (that is, sends them to the address from which the peer's RTP packets are coming).

ITSP Driven Distinctive Ringing

The device offers 10 ring and 10 call-waiting tone patterns in each ring profile. These patterns are numbered from 1 to 10. Each pattern also comes with a configurable name. You can assign a different default ring to each trunk on the device.

An ITSP can tell the device which ring pattern to use by name for a call routed to SP1/SP2 by inserting an Alert-Info header in the SIP INVITE sent to the device. The Alert-Info must include a URI. For example:

Alert-Info: <http://www.xyz.com/some-folder/bellcore-dr4>

When the device receives this, it looks for a ring tone name or call-waiting tone name in the ring profile that matches the Alert-Info URI. Ring tone names are compared case-insensitively. If a match is found, the device plays the corresponding ring or call-waiting tone. Otherwise, the device plays the default ring.

RTP Statistics – the X-RTP-Stat Header

When ending an established call, the device can include a summary of the RTP statistics collected during the call in the SIP BYE request or the 200 response to the SIP BYE request sent by the peer device. The summary is carried in an X-RTP-Stat header in the form of a comma-separated list of fields. The reported fields are:

- PS = Number of Packets Sent
- PR = Number of Packets Received
- OS = Number of bytes sent
- OR = Number of bytes received
- PL = Number of packets lost
- JI = Jitter in milliseconds
- LA = Decode latency or jitter buffer size in milliseconds
- DU = Call duration in seconds
- EN = Last Encoder Used
- DE = Last Decoder Used

For example:

X-RTP-Stat:PS=1234,OS=34560,PR=1236,OR=24720,JI=1,DU=1230,PL=0,EN=G711U, DE=G711U

To enable the X-RTP-Stat feature, set the **ITSP Profile X – SIP::X_InsertRTPStats** parameter to YES (or TRUE).

Media Loopback Service

The device supports the media loopback draft as described in *draft-mmusic-media-loopback-13.txt*. You can enable or disable this feature from **System Management > Device Admin > Media Loopback**.

The device supports the following media loopback features:

- Loopback modes: `loopback-source` and `loopback-mirror`
- Loopback types: `rtp-media-loopback` and `rtp-packet-loopback`
- Loopback packet formats: `encaprtp`, `loopbkprimer`

When the device acts as a loopback mirror, it always sends primer packets so that incoming packets can get through NAT/Firewall. The media loopback feature is controlled by the following parameters (in the **Device Admin – Media Loopback** section):

- **AcceptMediaLoopback** – Enable device to accept incoming call that requests media loopback. Default is YES.
- **MediaLoopbackAnswerDelay** – The delay in ms before the device answers a media loopback call. Default is 0.

- **MediaLoopbackMaxDuration** – The maximum duration to allow for an incoming media loopback call. Default is 0, which means the duration is unlimited.

The device rejects an incoming media loopback call if:

- Handset port is off-hook.
- Handset port is ringing.

The device terminates an inbound media loopback call already in progress when:

- Handset port is off-hook.
- Handset port is ringing.

To make an outgoing loopback call, dial one of the following star codes before dialing the target number:

- *03 – Make a Media loopback call.
- *04 – Make an RTP packet loopback call.

Note that outbound Media Loopback Call isn't subject to call duration limit. It lasts until you hang up or until the called device ends the call.

For more information on general ITSP parameters, see the [ITSP Profile X – General Web Page \(X = A, B, C, D, E, F, G, H\) Parameter Guide](#) table in the [Parameter Reference](#) section.

For more information on ITSP SIP settings parameters, see the [ITSP Profile X – SIP Web Page \(X = A, B, C, D, E, F, G, H\) Settings Parameter Guide](#) table in the [Parameter Reference](#) section.

For more information on ITSP RTP settings parameters, see the [ITSP Profile X – RTP Web Page \(X = A, B, C, D, E, F, G, H\) Parameter Guide](#) table in the [Parameter Reference](#) section.

Using SP_n as a Proxy for a SIP IP Phone

An SP service can be set up as a proxy for a legacy IP phone to let the phone access OBiTALK or OBiBlueTooth (on SP_n) installed on the device. This proxy mode of operation must be explicitly enabled in the SP's configuration on the device. It is disabled by default. The IP phone using this proxy service is known as the *local_client* of the SP service. It must be installed on the LAN side of the device.

In this mode, SP_n accepts SIP Registration from the client device from the LAN side, which must be using the same user-id and password as this SP_n's **AuthUserName** and **AuthPassword** parameters for authentication. This client device can also send SIP INVITE to the device at this SP to make calls. This SP's **InboundCallRoute** must be set up with the proper routing rule to handle calls from the *local_client*.

The SIP Proxy Server parameter on the client device must be sent to:

```
<obi-number>.pnn.obihai.com:<spn-user-agent-port>
```

where <obi-number> is the 9-digit OBi number of this device, and <spn-user-agent-port> is SP_n's **X_UserAgentPort** parameter.

For example, SP1 has a *local_client* with the user-id 4086578118. The client wishes to make and receive calls on SP3. The SP1 **InboundCallRoute** shall include the following rule:

```
{4086578118>:sp3}
```

The SP3 **InboundCallRoute** shall be: {sp1(408657118@local_client)}

For more information on SP_n services parameters, see the [SP_n Services \(n = 1, 2, 3, 4, 5, 6, 7, 8\) Settings](#) table in the [Parameter Reference](#) section.

OBiTALK Service Settings

For more information on OBiTALK service settings parameters, see the [OBiTALK Service Settings Parameter Guide](#) table in the [Parameter Reference](#) section.

For more information on OBiTALK calling features parameters, see the [OBiTALK Calling Features Parameter Guide](#) table in the [Parameter Reference](#) section.

Auto Attendant Service

The device call processing Auto Attendant (AA) invoked by including “aa” in the inbound call routing rule associates the interface on the device processing an incoming call. An OBi device supports only one session of AA at a time. Additional calls routed to the AA while a session is in progress are rejected by the AA as busy.

AA Callback Service

The device offers two methods for the AA to call you back at a number that you picked (or designated by the admin of the device).

The first method is by statically configuring a trunk's **InboundCallRoute**. A rule can be added to the **InboundCallRoute** parameter to have the AA call back the caller's or any other number, if the caller hangs up before the AA answers. The rule should indicate that *aa (callback-number)* is the target destination of the call, where *callback-number* is the number that the AA should call back if the caller hangs up before the AA answers the call. For example, the following rule:

```
{ (< **1 > (14089913313 | 12121559801)) : aa ($1) }
```

says that if 14089913313 or 12121559801 calls, the call is routed to AA. If the caller hangs up before the AA answers, AA calls the number represented by \$1. Recall that \$1 is expanded into the caller number after processing by the digit map on the left side of the colon. In this case, it is the caller's number prepended by **1. The **1 is required for outbound call routing when AA calls back. Here, it indicates SP1 is to be used for calling back (assuming default value of the AA **OutboundCallRoute** parameter).

The **AA Service::CallbackAnswerDelay** parameter controls the number of milliseconds before AA answers when a callback number is specified. The default value is 10000 ms (10 seconds). Without the *callback-number* argument, the AA behaves the normal way and the answer delay is governed by the **AA Service::AnswerDelay** parameter.

The second method is by selecting AA option 3 to “Enter a callback number” after the AA answers the call. The caller can explicitly enter the number to be called back by the AA. If a valid number is entered, AA says “Thank You” and “Goodbye”, and then starts calling back 2 seconds after the current call has ended. If number entered is invalid, AA plays a SIT tone followed by an error message. Note that the variable \$1 (representing the caller's number) is carried over to the subsequent AA callback call. The AA **DigitMap** can include \$1 to be used in a callback context. For example, the following rule in the AA **DigitMap**

```
(< 00 : **1 $1 > | ... )
```

says that if the AA dials 00, the device transforms it into the caller's number prepended by **1. In other words, if the caller wants the AA to call back the current number (typically the case), he can simple enter 00# after selecting option 3 on the AA menu. Note that \$1 can only be used as part of a substitution element in the digit map. It must not be used for matching elements since its value is unknown.

User-Recorded Prompts

The device supports 10 user-recordable prompts, which are referred to as the *User1* to *User10* prompts. See the [Customized AA Prompts Backup and Restore](#) section for more information on how they can be duplicated from one device onto another device.

Customizing AA Prompt Lists

AA doesn't play individual user prompts directly. Instead it plays a comma-separated list of prompt elements, known as a *Prompt List*. A prompt element can be a user prompt with optional parameters, or a control element. A user prompt is referred as `%User<N>%` where `<N>` = 1 – 10. In a prompt list this can be followed by a `;r=<start>-<end>` parameter that specifies the range to play for that prompt, where:

`<start>` = starting time mark in milliseconds. 0 is the default if `<start>` is omitted.

`<end>` = ending time mark in milliseconds. The end of the prompt is the default if `<end>` is omitted.

If the `r=` parameter is omitted, the full range of the prompt is played.

Examples:

`%User1%;r=1000` means play the *User1* prompt starting at 1000 ms mark to the end.

`%User2%` means play the entire *User2* prompt from start to finish.

`%User3%;r =1300-3720` means play the *User3* prompt starting from the 1300 ms mark to the 3720 ms mark.

`%User4%;r=3200-1200` means don't play anything since `<end>` is less than `<start>`.

Each prompt list control element starts with an ampersand '&' in a prompt list. The following control elements are supported: `&pause(<duration>)` means pause playing for a number of seconds as given by the `<duration>` parameter.

An example prompt list:

`%User1%;r=105,&pause(3),%User5%,%User9%;r=0-1350,&pause(15)`

You can replace any of the following AA prompt lists with your own specified prompt lists:

Automated Attendant Prompt Lists

AA Prompt List	System Default	Prompt to Be Played
Welcome	Welcome to OBi Attendant.	Once, at the beginning when the AA starts.
InvalidPin	Invalid PIN.	After user enters an invalid PIN.
EnterPin	Enter PIN.	Prompts user to enter a valid PIN.
MenuTitle	Main Menu.	Once, after Welcome and before announcing the menu options.
Menu	Press 1 to continue this call. Press 2 to make a new all. Press 3 to enter a callback number.	A couple of times after MenuTitle.
PleaseWait	Please wait while your call is being connected.	Once, after user enters a phone number to call.

Automated Attendant Prompt Lists

AA Prompt List	System Default	Prompt to Be Played
EnterNumber	Enter number followed by the # key.	Prompts you to enter a valid number after you select option 2 or option 3.
Bye	Thank you for choosing Obihai Technology. Goodbye.	When you press * or # key to leave the AA.

Voice Gateways

A voice gateway (VG) in this context is another OBi device that lets incoming OBiTALK callers to call farther on one or more of its trunks (such as SP1). The caller can call the gateway first with a normal OBiTALK call, get the AA, and then dial the target number. For authentication, the AA may ask you to enter a PIN before establishing the second call. This way of dialing is known as 2-stage dialing.

On the other hand, a gateway can be configured on the originating device such that the caller can dial the target number directly without going through the AA. This method of dialing is called direct dialing or 1-stage dialing. Since it isn't possible to enter a PIN in the case of direct dialing, a userid/password pair can be configured for the gateway also so that the device can authenticate with the gateway automatically using HTTP digest method. HTTP digest authentication is optional. You don't need to provide user/password if the gateway doesn't require authentication for direct dialing.

The device allows you to specify as many as eight gateways. Each gateway is addressed using its factory-assigned OBi Number. A gateway is conceptually a trunk with its own **DigitMap**. You can refer to a gateway and its associated **DigitMap** with the short trunk names **VGn** and **(Mvgn)**, for $n = 1, 2, 3, \dots, 8$. **VGn** and **(Mvgn)** can be used in call routing rules and digit maps just like other real trunks.

As an example, you can add the rule `{(1xxx xxx xxxx):vg2}` in the Phone port's **OutboundCallRoute** to let the device dials out using VGs when caller dials any 11-digit number starting with 1. On the gateway side, you can add the corresponding rule `{>(1 xxx xxx xxxx):sp1}` in the **OBiTALK Service::InboundCallRoute** parameter to make the call on its SP1 trunk. You can change the last rule to `{(290 333 100|200 444 101)>(1 xxx xxx xxxx):sp1}` if you want to limit the gateway to allow just the two stated caller numbers to make such calls.

Starting with firmware release 1.2, a gateway may also be configured with a SIP URL as the access number to be called by the device over one of the SP trunks. For example, one can set the gateway access number as **SP1(some-sip-server.mydomain.com)**, or **SP2(192.168.15.111:5062)**.

Note that when using an SP trunk to access a SIP gateway, the device:

- Won't use the outbound proxy, ICE, or STUN, regardless of the settings on the SP trunk.
- Uses only the device's local address as the SIP Contact, and ignores any NATed address discovered by the device.
- Uses the gateway's SIP URL to form the FROM header of the outbound INVITE.
- Uses the gateway's **AuthUserID** and **AuthPassword** for authentication.
- Applies the symmetric RTP concept.

For more information on Voice Gateway parameters, see the [Gateways Parameter Guide](#) table in the [Parameter Reference](#) section.

Trunk Groups

As the name implies, a trunk group is a group of trunks. If a call is routed to a trunk group, the device picks one of the available trunks from the group to make the call. Availability of trunk is based on:

- Whether the trunk's digit map allows the number to call, AND
- Whether the trunk has capacity to make one more call

As many as four trunk groups can be configured on a device. Each trunk group is conceptually another trunk with its own **DigitMap**. A trunk group and its associated **DigitMap** are referenced using the short names **TG n** and **(Mtgn)**, where $n = 1, 2, 3, 4$. They can be referenced in other digit maps and call routing rules so that calls can be routed to a particular trunk group.

Only trunks can be added to a trunk group. These include: PP1, SP1, SP2, ..., SP9, VG1, VG2, ..., VG8, TG1, TG2, ... TG4. Note that a TG can include another TG (that is, TG can be recursive). However, you must make sure this doesn't result in infinite recursion.

For more information on trunk group parameters, see the [Gateways Parameter Guide](#) table in the [Parameter Reference](#) section.

Parameter Reference

This section lists the ATA parameters in the order shown on the system web interface.

Depending on your device or your settings, the system web interface may not present all of these parameters to you.

Status Parameters

The Status Parameters web pages show read-only values for certain parameters on your device. They include these pages:

- [System Status Settings](#)
- [Call Status Settings](#)
- [Call History](#)
- [SP Services Stats Settings](#)

System Status Settings

The System Status page displays the status of your device and the configured services. It also displays the device product information.

System Status Settings

Parameter	Description
<i>WAN Status (DeviceInfo.Network.Status.WAN.)</i>	
AddressingType	Method currently used by the phone to get an IP address assignment. Example value: DHCP
IPAddress	IP address currently assigned to the phone when using static IP addressing. Example value: 192.168.15.165
SubnetMask	Subnet mask to use when using static IP addressing. Example value: 255.255.255.0
DefaultGateway	Gateway to use when using static IP addressing. Example value: 192.168.15.1
DNSServer1	URL for domain name server 1 when using static IP addressing. Example value: 8.4.4.4

System Status Settings

Parameter	Description
DNSServer2	URL for domain name server 2 when using static IP addressing. Example value: 4.2.2.2
MACAddress	MAC address installed on the phone. Example value: 9CADEF90004E
LLDP-MEDStatus	Enables LLDP media endpoint discovery for improved network connections. Example value: Enabled

Product Information (DeviceInfo.)

ModelName	Your device's model name. Example value: OBi508
MACAddress	Your device's MAC address. Example value: 9CADEF90004E
SerialNumber	Your device's serial number. Example value: 88H01NA00ZXV
OBiNumber	Your device's OBi number, a value that uniquely identifies your phone to Polycom and to other OBi devices. Example value: 552 860 300
HardwareVersion	Your device's hardware version. Example value: 1.1
SoftwareVersion	Your device's installed software version. This value changes with a firmware update or downgrade. Example value: 6.3.0.15058
SystemTime	Shows the current time on the system. Example value: 15:32:35 01/29/2019, Wednesday
UpTime	With last Reboot Reason in parentheses. Example value: 20 Days 5:04:13 (2)
CertificateStatus	Indicates if a device certificate is installed on the device. Example value: Installed
CustomizationStatus	Indicates if this device is a customized unit. Example value: Generic

OBiBT OBiBT Bluetooth Adapter Status (VoiceService.1.X_BT.1.Stats.)

Status	OBiBT Bluetooth adapter device state.
Discoverable	Time duration the OBiBT Bluetooth adapter remains discoverable.
CallState	Current call state. Example value: 0 Active Calls
BindingService	The service this OBiBT Bluetooth binds to.

System Status Settings

Parameter	Description
<i>SPn Service Status (VoiceService.1.VoiceProfile.1.Line.n.), n = 1 – 8</i>	
Status	Registration status of this service. If there are problems with the registration or authentication, the SIP 4xx – 6xx error code and error message display here. This is useful information for troubleshooting issues with SIP-based services. Example value: Registered (server=192.168.15.118; expire in 39s)
PrimaryProxyServer	IP address of the current primary proxy server if proxy server redundancy is enabled on this service. Example value: 10.100.123.234
SecondaryProxyServer	IP address of the current secondary proxy server if proxy server redundancy and secondary registration are both enabled on this service. Example value: 10.100.234.123
CallState	Describes the state of an active call on this service. Example value: 0 Active Calls
<i>OBiTALK Service Status (VoiceService.1.X_P2P.1.Stats.)</i>	
Status	Connection status with the OBiTALK network. Example value: Normal (User Mode)
CallState	Describes the state of an active call on OBiTALK. Example value: 0 Active Calls

Call Status Settings

The Call Status page shows a number of running call statistics and state parameters for each active call. The call status is only available during the lifetime of the call.

Call Status Descriptions

Status	Description
Peer Name	Call peer's name.
Peer Number	Call peer's number.
Start Time	Starting time of the call.
Duration	Duration of the call.
Peer RTP Address	The peer address and port where RTP packets are sent to.
Local RTP Address	The local address and port where RTP packets are sent from.
RTP Transport	The transport used for RTP (UDP, TCP, or SSL).
Audio Codec	The audio encoder and decoder being used for this call.
RTP Packetization (ms)	The transmitted and received packet sizes in milliseconds.

Call Status Descriptions

Status	Description
RTP Packet Count	Total number of RTP packets transmitted and received.
RTP Byte Count	Total number of RTP bytes transmitted and received.
Peer Clock Differential Rate	Clock difference between this phone and the peer in ppm (parts per million).
Packets Out-of-Order	Number of received RTP packets that are out of order.
Packets Lost	Number of incoming RTP packets assumed lost.
Packet Loss Rate	Amount of incoming RTP packets assumed lost rate in percent.
Packet Drop Rate	Amount of incoming RTP packets dropped in percent.
Jitter Buffer Length	Size of the current jitter buffer in milliseconds.
Received Interarrival Jitter	Average measured network jitter in the received direction in milliseconds.
Max Interarrival Jitter	Maximum measured network jitter in the received direction in milliseconds.
Jitter Buffer Underruns	Amount of jitter buffer underruns during the call.

For each entry on the call status page, the following buttons are available:

- **Remove:** This button is available for all calls. Pressing this button ends that call.
- **Record:** Pressing this button lets you record the current conversation in an audio (.au) file.

Call History

The Call History page shows the last 200 calls. Detailed call information is available, including what terminals were involved, the name (if available) of the peer endpoints making the call and the direction / path the call took, and the time events took place.

The following buttons are available:

- **Remove All:** Clicking this button erases the entire call history.
- **Save All:** Clicking this button saves the call history to the `callhistory.xml` file.

SP Services Stats Settings

You can find statistics relevant to SP_n on the SP_n Stats page, where $n = 1-8$.

Services, Phone, and Line Status Parameter Guide

Parameter	Description
ResetStatistics	
ResetStatistics	This is a Boolean option to reset the statistics for this SP Service. After submitting this change, the value reverts automatically to the default value. Default setting: <code>false</code>

Services, Phone, and Line Status Parameter Guide

Parameter	Description
<i>RTP Statistics</i>	
PacketsSent	Total RTP packets sent on this line.
PacketsReceived	Total RTP packets received on this line.
BytesSent	RTP payload bytes sent for this line.
BytesReceived	RTP payload bytes received for this line.
PacketsLost	Number of RTP packets lost on this line.
Overruns	Number of jitter buffer overruns on this line.
Underruns	Number of jitter buffer underruns on this line.

System Management Parameters

Phone Port Status Parameter Guide

Parameter	Description
<i>Port Status</i>	
State	Port status, such as on-hook, off-hook, ringing.
LoopCurrent	Loop current in mA.
VBAT	Phone port battery voltage in volts. Not applicable for LINE port.
TipRingVoltage	Sensed differential Tip/Ring voltage in volts.
LastCallerInfo	Caller ID of previous call.

The System Management parameter web pages show network parameters on your device. They include these pages:

- [WAN Settings](#)
- [Auto Provisioning Settings](#)
- [Device Admin Settings](#)
- [Device Update](#)

WAN Settings

This page lists the Ethernet settings for your device.

WAN Settings

Parameter	Description	Default Setting
<i>Internet Settings (DeviceInfo.WAN)</i>		
AddressingType	Method currently used by the handset to get an IP address assignment. Example value: DHCP	DHCP
IPAddress	IP address currently assigned to the handset when using static IP addressing. Example value: 192.168.15.165	
SubnetMask	Subnet mask to use when using static IP addressing. Example value: 255.255.255.0	255.255.255.0
DefaultGateway	Gateway to use when using static IP addressing. Example value: 192.168.15.1	
DNSServer1	URL for domain name server 1 when using static IP addressing. Example value: 8.4.4.4	
DNServer2	URL for domain name server 2 when using static IP addressing. Example value: 4.2.2.2	
VLANEnable	VLAN Operation Bool.	
VLANID	Valid range is 0 - 4094 (4095 is reserved). 0 means VLAN is disabled and egress packets are not tagged by the device. This setting applies to all packets sent by the device.	0
VLANPriority	Valid choices are 0 - 7. This setting applies to all packets sent by the device.	0
VLANDiscovery	Enables taking VLAN setting from DHCP options. Choice of: - Disabled - Fixed - Custom	Disabled
VLANDiscoveryOption	Specifies which DHCP option to use for VLAN discovery.	129
802_1XMode	Port-based network access control provides an authentication mechanism to attach to a LAN. Choice of: - Disable - MDS - TLS - TTLS-MSCHAPv2 - PEAP-MSCHAPv2	Disable
802_1XIdentity	User name for 802.1x authentication.	

WAN Settings

Parameter	Description	Default Setting
802_1XPassword	Password for 802.1x authentication.	
802_1XTLSecurityProfile	The TLS platform profile to use for device certification.	1
LLDP-MED	Enable LLDP-MED discovery.	
LLDP-MEDExclusivePeriod	Delay in seconds before getting or setting up an IP address based on AddressingType to exclusively perform LLDP-MED.	5
LLDP-MEDAssetID	The LLDP-MED Asset ID to broadcast during LLDP-MED discovery.	\$OBN
OpenSSLCiphers	OpenSSL ciphers to support for all SSL/TLS connections. An empty value tells the device to use the default ciphers. A valid value must start with <code>DEFAULT:</code> or <code>HIGH:</code> .	Yes

Local Time (DeviceInfo.Time.)

CurrentLocalTime	Current local date and time of the device (read-only parameter).	
CurrentNTPServer1	Host name or IP address of the current first NTP server.	None
CurrentNTPServer2	Host name or IP address of the current second NTP server.	None

Time Service Settings (DeviceInfo.Time.)

NTPServer1	Host name or IP address of the first NTP server.	pool.ntp.org
NTPServer2	Host name or IP address of the second NTP server.	
LocalTimeZone	Local time zone.	GMT-8:00 (Pacific Time)
DaylightSavingTimeEnable	Enables daylight saving time on the unit.	Yes
DaylightSavingTimeStart	Daylight Saving Time Start Date Format: month/day/week/hh:mm:ss, where month = 1-12, day = 1-21, weekday=0, 1-7 (0=special), hh = 0-23, mm = 0-59, ss = 0-59. If weekday = 0, daylight saving starts on the given month/day. Otherwise, it starts on the weekday on or after the given month/day if day > 0, or on the weekday on or before the last-day-of-the-given-month+day+1 (note that day = -1 equivalent to last day of the month). Seconds can be omitted if the value is 0. Minutes and seconds can be omitted if both values are 0.	3/8/7/2
DaylightSavingTimeEnd	Daylight Saving Time End Date. Same format as DaylightSavingTimeStart .	11/1/7/2

WAN Settings

Parameter	Description	Default Setting
DaylightSavingTimeDiff	Amount of time to add to current time during Daylight Saving Time. Format: [-]hh:mm:ss. Seconds can be omitted if the value is 0. Minutes and seconds can be omitted if both values are 0.	1

DHCP Client Settings (X_DHCPClient.)

ExtraOptions	Comma-separated list of extra DHCP options to be requested. Choice of: • 42 • 66 • 150 • 159 • 160 • 161	66, 42
---------------------	---	--------

DNS Control (X_DNSControl.)

DNSQueryOrder	When more than one DNS servers are available, the unit attempts to resolve a domain name by querying each server sequentially until a successful result is received. The parameter controls the order in querying the servers. Choose from: • DNS Server1, DNS Server2, DHCP Offered DNS Servers • DHCP Offered DNS Servers, DNS Server1, DNS Server2	DNS Server1, DNS Server2, DHCP Offered DNS Servers
DNSQueryDelay	When multiple DNS servers are available, the unit attempts to resolve a domain name by querying each server sequentially until a successful result is received. This parameter controls the number of seconds between successive DNS queries made by the unit for a given domain name. Choose from 0 to 5 seconds.	2

WAN Settings

Parameter	Description	Default Setting
Local DNS Records (X_LocalDNSRec.)		
N, where N = 1-32	One of 32 Local DNS Records (numbered 1 – 32). Each record is a mini script of either of the following formats: - Name=A, A, A, . . . - Name=R, R, R, . . . where Name represents the domain name to be resolved locally, and has the format <code>prefix+domain</code> (such as <code>machine.sip+obiwai.com</code>). Everything after '+' is considered as the domain to be appended to the host field in each R on the right hand side. '+' is optional. If missing, the full domain must be used in every R. A represents an A record, which is just an IP address, such as 192.168.12.17. R represents an SRV record and has the format: <code>{host:port,pri,wt}</code> where - host is a host name with or without domain part (such as xyz, xyz.abc.com.). A dot (.) at the end of host indicates it is a complete host name that does not require the domain to be appended. - port is a port number (such as 5060). port is optional. The default to use is based on the protocol (5060 for SIP, 80 for HTTP, and so forth). - pri is the priority. Valid value is 0 (highest) – 65535 (lowest). - wt is the weight. Valid value is 0 (lowest) – 65535 (highest). wt is optional. 1 is the default if not specified. pri is optional only if wt isn't specified. 1 is the default if not specified. The enclosing curly brackets { } are also optional if there is only one R, or if no comma appears inside the R. Examples: <pre>_sip. udp+obiwai.com=abc,xyz,pqr:5080,{mmm,2},{super.abc.com.} abc.obiwai.com=192.168.15.118,192.168.15.108</pre> Note: If the A record of a given host name cannot be found in any of the local DNS records, the handset attempts to resolve it using external DNS queries. Any change applied to a local DNS record needs a reboot to take effect.	None

Auto Provisioning Settings

The Auto Provisioning web page shows all the parameters related to remote provisioning of the device.

Auto Provisioning Parameter Guide

Parameter	Description	Default Setting
<i>Auto Firmware Update (X_DeviceManagement.FirmwareUpdate.)</i>		
Method	Current operational method of auto firmware updating. Choose from: - Disabled: don't check for f/w upgrade from FirmwareURL. - System Start: Check for f/w upgrade from FirmwareURL just once on system start. - Periodically: Check for f/w upgrade from FirmwareURL on system start, and then periodically at the interval specified in the Interval parameter. - Time of Day: Check once at the given TimeOfDay value. Note: The first firmware upgrade check on system start is performed after a random delay of 0 to 30 seconds.	Disabled
Interval	When Method is set to Periodically , this is the number of seconds between each checking of f/w upgrade check from FirmwareURL . If value is 0, the device checks once only on system start (equivalent to setting Method to System Start).	0
TimeOfDay	Time of the day in hh:mm[+rr] format, valid when method is set to Time of Day .	00:00+30
RandomDelayRange	The range of delay in seconds inserted before the first attempt only. The minimum value is 0.	30
FirmwareURL	URL of firmware package. URL must include scheme. Supported schemes are http:// and tftp://	
TLSSecurityProfile	Security profile when using HTTPS. Choices are 1 or 2.	1
DnsLookupType	Controls what type of DNS record to lookup. Choose from: - A Record Only - SRV Record Only - Try Both	A Record Only
DnsSrvPrefix	Controls whether to add a standard prefix to the domain name when looking up a SRV Record. For HTTP and HTTPS, the prefix to add is _http._tcp. . For TFTP, the prefix to add is _tfto._udp. Choose from: - No Prefix - With Prefix - Try Both	No Prefix
Username	Username for authentication, if needed, if scheme is http://	
Password	Password for authentication, if needed, if scheme is http://	
Suspend	Suspend firmware update until canceled.	false

ITSP Provisioning (X_DeviceManagement.ITSPProvisioning.)

Auto Provisioning Parameter Guide

Parameter	Description	Default Setting
Method	Current operational method of Provisioning. Choose from: - Disabled: don't download from ConfigURL. - System Start: Download from ConfigURL just once on system start. - Periodically: Download from ConfigURL on system start, and then periodically at the interval specified in the Interval parameter. - Time of Day: Check once at the given TimeOfDay value. Note: First download on system start is performed after a random delay of 30 to 90 seconds if there is a firmware update scheduled at the beginning, or a random delay of 10 to 70 seconds.	System Start
Interval	When Method is set to Periodically, this is the number of seconds between download from ConfigURL . If value is 0, device downloads once only on system start (equivalent to setting Method to System Start).	0
TimeOfDay	Time of the day in hh:mm[+rr] format, valid when method is set to Time of Day.	00:00+30
ConfigURL	URL of config file.	tftp://\$DHC POPT66/\$MAC .xml
DnsLookupType	Controls what type of DNS record to lookup. Choose from: - A Record Only - SRV Record Only - Try Both	A Record Only
DnsSrvPrefix	Controls whether to add a standard prefix to the domain name when looking up a SRV Record. For HTTP and HTTPS, the prefix to add is _http._tcp.. For TFTP, the prefix to add is _tfto._udp.. Choose from: - No Prefix - With Prefix - Try Both	No Prefix
Override	Defines which local settings can be overridden by this provisioning. Choose from: - All - All except user settings	All
GPRM0 to GPRM7	Non-volatile generic parameters that can be referenced in other parameters, such as ConfigURL .	
TPRM0 to TPRM3	Temporary variables used in scripts for ConfigURL .	

OBitALK Provisioning (X_DeviceManagement.Provisioning.)

Auto Provisioning Parameter Guide

Parameter	Description	Default Setting
Method	Current operational method of Provisioning. Choose from: - Disabled: Don't download from ConfigURL. - System Start: Download from ConfigURL just once on system start. - Periodically: Download from ConfigURL on system start, and then periodically at the interval specified in the Interval parameter. - Time of Day: Check once at the given TimeOfDay value. Note: First download on system start is performed after a random delay of 30 to 90 seconds if there is a firmware update scheduled at the beginning, or a random delay of 10 to 70 seconds.	System Start
Interval	When Method is set to Periodically, this is the number of seconds between download from ConfigURL . If value is 0, device downloads once only on system start (equivalent to setting Method to System Start).	0
TimeOfDay	Time of the day in hh:mm[+rr] format, valid when method is set to Time of Day.	00:00+30
ConfigURL	URL of config file.	
DnsLookupType	Controls what type of DNS record to lookup. Choose from: - A Record Only - SRV Record Only - Try Both	A Record Only
DnsSrvPrefix	Controls whether to add a standard prefix to the domain name when looking up an SRV Record. For HTTP and HTTPS, the prefix to add is <code>_http._tcp..</code> For TFTP, the prefix to add is <code>_tfto._udp..</code> Choose from: - No Prefix - With Prefix - Try Both	No Prefix
GPRM0 to GPRM7	Non-volatile generic parameters that can be referenced in other parameters, such as ConfigURL .	
TPRM0 to TPRM3	Temporary variables used in scripts for ConfigURL .	
User-Defined Macro 0–3 (\$UDM0 – \$UDM3)		
Value	The value can be any plain text or a valid canonical parameter name preceded by a \$ sign. For example: \$X_DeviceManagement.WebServer.Port Note: Here you MUST NOT enclose the parameter name following the \$ sign with braces or parentheses.	

Auto Provisioning Parameter Guide

Parameter	Description	Default Setting
ExpandIn	This is a comma-separated list of canonical parameter names, where the macro expansion can be used. As many as three parameter names can be specified. Specify ANY to allow the macro to expand in any parameter. Example: <pre>X_DeviceManagement.HTTPClient.UserAgent</pre> Note: There is no \$ sign in front of the parameter name. The macro can't be used in any parameter value if this value is set to blank (the default).	
SyntaxCheckResult	This is read only status value regarding the syntax of the UDM. Pass means that this UDM is valid. Otherwise, it shows the syntax error detected by the device either in the Value or ExpandIn parameters of the UDM.	

WAN Settings Parameter Guide

Parameter	Description	Default Setting
<i>Internet Settings (DeviceInfo.WAN)</i>		
AddressingType	The method to assign the IP address, subnet mask, and default gateway to the device. Choose from: DHCP: Settings are assigned by DHCP Server Static: Settings are taken from the manually configured values. PPPoE: Settings are acquired by PPPoE Protocol.	DHCP
IPAddress	IPv4 address to assign to the device when AddressingType is set to Static .	
SubnetMask	Subnet mask to use when AddressingType is set to Static .	
DefaultGateway	Default gateway IP address to assign to the device when AddressingType is set to Static .	
DNSServer1	IP address of the first DNS server to use, in addition to the ones obtained from the DHCP server when DHCP is also enabled. If AddressingType is set to Static, the device only uses DNSServer1 and DNSServer2 for DNS lookup. It tries as many as five DNS servers when attempting to resolve a domain name. DNSServer1 and DNSServer2 are tried first, whichever is specified, and then the addresses obtained from the DHCP Server if available.	

WAN Settings Parameter Guide

Parameter	Description	Default Setting
DNSServer2	IP address of the second DNS server to use, in addition to the ones obtained from the DHCP server when DHCP is also enabled. If AddressingType is set to Static, the device only uses DNSServer1 and DNSServer2 for DNS lookup. It tries as many as five DNS servers when attempting to resolve a domain name. DNSServer1 and DNSServer2 are tried first, whichever is specified, and then the addresses obtained from the DHCP Server if available.	
MTUSize	MTU size in number of bytes.	1500
MACAddressClone	The MAC Address to clone (instead of using the factory installed MAC Address).	
PPPoEACName	PPPoE access concentrator name. Enter if it is required.	
PPPoEServiceName	PPPoE service name. Enter if it is required String.	
PPPoEUsername	PPPoE account username provided by your ISP String.	
PPPoEPassword	PPPoE account password String.	
PPPoEKeepAlive	PPPoE keep alive period in seconds.	60
VLANEnable Enable	VLAN Operation Bool.	false
VLANID	Valid range is 0 – 4094 (4095 is reserved). 0 means VLAN is disabled and egress packets are not tagged by the device. This setting applies to all packets sent by the device.	0
VLANPriority	Valid choices are 0 – 7. This setting applies to all packets sent by the device.	
LLDP-MED	Enable LLDP-MED Discovery.	false
LLDPMEDExclusivePeriod	Delay in seconds before getting or setting up an IP address based on AddressingType to exclusively perform LLDP-MED.	
LLDP-MEDAssetID	The LLDP-MED Asset ID to broadcast during LLDP-MED Discovery.	\$OBN
Local Time		
CurrentLocalTime	Current local date and time of the device (read-only parameter).	
Time Service Settings		
NTPServer1	Host name or IP address of the first NTP server.	pool.ntp.org
NTPServer2	Host name or IP address of the second NTP server.	
LocalTimeZone	Local time zone.	GMT-08:00 (Pacific Time)

WAN Settings Parameter Guide

Parameter	Description	Default Setting
DaylightSavingTimeEnable	Enables daylight saving time on the unit.	Yes
DaylightSavingTimeStart	Daylight Saving Time Start Date. Format: month/day/weekday/hh:mm:ss, where month=1-12, day=±(1-31), weekday=0,1-7 (0=special, 1=Monday, 7=Sunday), hh=0-23, mm=0-59, ss=0-59. If weekday=0, daylight saving starts on the given month/day. Otherwise, it starts on the weekday on or after the given month/day if day > 0, or on the weekday on or before the last-day-of-given-month+day+1 (note that day = -1 equivalent to last day of the month). :ss can be omitted if the value is 0. :mm:ss can be omitted if mm and ss are both 0.	3/8/7/2
DaylightSavingTimeEnd	Daylight Saving Time End Date. Same format as Start Date.	11/1/7/2
DaylightSavingTimeDiff	Amount of time to add to current time during Daylight Saving Time. Format: [-]hh:mm:ss. :ss can be omitted if it is 0. :mm:ss can be omitted if both are 0.	1

WAN Failover Control (DeviceInfo.WAN.FailoverControl.)

Mode	Method to use to drive the failover from Ethernet to LTE. Choices are: • Disable • Ping • DNS Query	Disable
Targets	A comma-separated list of servers to use to test for failover.	8.8.8.8, 8.8.4.4, 4.2.2.2, 209.244.0.3
Interval	Failover testing interval in seconds.	30
Threshold	Number of consecutive negative test results to trigger failover.	3
PrimaryFailoverDelay	Delay in seconds before restarting test for failover after failing back to the primary interface (Ethernet)	300
SecondaryFailoverDelay	Delay in seconds before restarting test for failing back after failover to the secondary interface (LTE)	300
KeepInterfaceAwakePeriod	Interval to send keep alive traffic on the LTE interface to keep it awake. Set the value to 0 to disable this keep alive feature.	60

DNS Control

WAN Settings Parameter Guide

Parameter	Description	Default Setting
DNSQueryOrder	When more than one DNS servers are available, the unit attempts to resolve a domain name by querying each server sequentially until a successful result is received. The parameter controls the order in querying the servers. Choose from: • DNS Server1, DNS Server2, DHCP Offered DNS Servers • DHCP Offered DNS Servers, DNS Server1, DNS Server2	DNS Server1, DNS Server2, DHCP Offered DNS Servers
DNSQueryDelay	When multiple DNS servers are available, the unit attempts to resolve a domain name by querying each server sequentially until a successful result is received. This parameter controls the number of seconds between successive DNS query made by the unit for a given domain name. Choose from 0 to 5 seconds.	2

WAN Settings Parameter Guide

Parameter	Description	Default Setting
Local DNS Records		
N where N = 1 – 32	One of 32 Local DNS Records (numbered 1 – 32). Each record is a mini script of the following format: <i>Name</i>=A,A,A,... or <i>Name</i>=R,R,R,... where <i>Name</i> represents the domain name to be resolved locally, and has the format <code>prefix+domain</code> (such as <code>machine.sip+obiwai.com</code>). Everything after '+' is considered as the domain to be appended to the host field in each <i>R</i> on the right hand side. '+' is optional; if missing, the full domain must be used in every <i>R</i>. <i>A</i> represents an <i>A</i> record that is just an IP address, such as <code>192.168.12.17</code>. <i>R</i> represents an SRV record and has the format: <code>{host:port,pri,wt}</code> where <i>host</i> is a host name with or without domain part (such as <code>xyz</code>, <code>xyz.abc.com.</code>). A dot (.) at the end of host indicates it is a complete host name that doesn't require the domain to be appended. <i>port</i> is a port number (such as 5060) <i>pri</i> is the priority. Valid value is 0 (highest) – 65535 (lowest) <i>wt</i> is the weight. Valid value is 0 (lowest) – 65535 (highest). <i>wt</i> is optional. If not specified, the default is 1. <i>pri</i> is optional only if <i>wt</i> isn't specified. 1 is the default if not specified. <i>port</i> is optional. The default to use is based on the protocol (5060 for SIP, 80 for HTTP, and so forth). The enclosing curly braces { } are also optional if there is only one <i>R</i>; or if there is no comma used inside the <i>R</i>. Examples: <code>_sip._udp+obiwai.com=abc,xyz,pqr:5080,{mmmm,2},{super.abc.com.}</code> <code>abc.obiwai.com=192.168.15.118,192.168.15.108</code> Note: If the <i>A</i> record of a given host name can't be found in any of the Local DNS Records, the device attempts to resolve it using external DNS queries. Any change applied to Local DNS Record needs a reboot to take effect.	

LAN Settings Parameter Guide

Parameter	Description	Default Setting
LAN Settings		
CurrentRouterIPAddress	The current IP address of the router on the LAN side (read only). It is blank if the device is operating in bridge mode.	
OperationMode	The Networking Operation Mode for the device. It can be one of the following values: - Router - Bridge	Router
RouterIPAddress	The LAN side IP address to be used by the router. If it conflicts with the WAN side IP address, the device automatically picks a different LAN side IP address to resolve the conflict.	192.168.10.1
SubnetMask	The LAN side Subnet Mask to be used by the router. It can be one of the following values: 255.255.255.0 255.255.255.128 255.255.255.192 255.255.255.224 255.255.255.240 255.255.255.248	255.255.255.0
DHCP Server Settings		
Enable	Enables the DHCP Server on the LAN side.	Yes
ClientAddressRangeStart	Together with the CurrentRouterIPAddress and SubnetMask parameters, this parameter determines the starting IP address to assign to DHCP clients. The value of this parameter is the starting value of the lower bits of the 32-bit starting IP address not masked by the SubnetMask , and it MUST fit within the unmasked range of the SubnetMask . Here are some examples:	100
SubnetMask	CurrentRouter IPAddress	ClientAddress RangeStart
255.255.255.0	192.168.10.1	100
255.255.255.0	192.168.2.1	50
		First Client IP Address
		192.168.10.100
		192.168.2.50
MaximumClients	The size of the IP address range from which to pick addresses to assign to DHCP clients that are not in the DHCP reservation list. If the range extends to addresses outside of the SubnetMask , a red exclamation mark (!) is shown next to the value on the device web page.	50
AddressLeaseTime	IP address lease time in minutes.	1440
LocalDomainName	Local Domain Name for the LAN.	

DHCP Reservation Parameter Guide

Parameter	Description	Default Setting
<i>(Reservation) 1 – 20</i>		
Enable	Enables this reservation.	No
Name	An optional name for easy identification of the client.	
MACAddress	Client MAC address in the format <code>xx:xx:xx:xx:xx:xx</code> where each <code>x</code> is a hex digit that can be in upper or lower case.	
IPAddress	The fourth (least significant) bytes of the IP address to be reserved. The upper three bytes of the IP address is determined by the router's LAN side subnet.	

Firewall and DMZ Parameter Guide

Parameter	Description	Default Setting
<i>DMZ Settings</i>		
Enable	Enables DMZ Service.	No
HostIPAddress	The IP address of the DMZ server.	
<i>Firewall Settings</i>		
Enable	Enables the firewall.	No
NATRedirection	Supports NAT Redirection, also known as NAT Loopback or Hairpin. This setting takes effect only if the firewall is enabled. Otherwise, this feature is disabled.	No
DRDOSAttackProtection	Enables protection against Distributed Reflection Denial of Service. This setting takes effect only if the firewall is enabled. Otherwise, this feature is disabled.	No
VPNPassthrough	Allows VPN (L2TP, PPTP, and IPSEC) traffic to pass through if enabled. Otherwise, all VPN traffic is blocked. This setting takes effect only if the firewall is enabled. Otherwise, this feature is <i>enabled</i> .	Yes

Port Forwarding Parameter Guide

Parameter	Description	Default Setting
<i>Port Forwarding n (DeviceInfo.LAN.PortForward.Rule.n.) for $n = 1, 2, \dots, 20$</i>		
Enable	Enables this port forwarding rule.	
RuleDescription	Description of this rule.	

Port Forwarding Parameter Guide

Parameter	Description	Default Setting
Protocol	The transport protocol for the specified port range in this rule. It can take one of the following values: - TCP: TCP only - UDP: UDP only - Both: TCP and UDP	TCP
StartingPortNumber	Starting port number of the forwarded port range.	
EndingPortNumber	Ending port number of the forwarded port range. If it is the same as StartingPort , only the one port equal to the StartingPort is forwarded.	
ServerIPAddress	The LAN side IP address to forward the packet to when it is received at a port on the WAN side within the port range in this rule with matching transport protocol.	
DestPortNumber	The port number on the LAN side host to which the traffic should be forwarded based on this rule.	

QoS Parameter Guide

Parameter	Description	Default Setting
QoS General Settings (DeviceInfo.Qos.BandwidthControl.)		
Enable	Enables QoS Service (take effect in router mode only).	No
UpStreamBandwidth	The total upstream bandwidth in kbps.	2048
RestrictedBandwidth	The bandwidth limit in kbps allocated to the Restricted Class when using the Primary (Ethernet) interface. The Restricted Class has the higher priority than any other priority classes. This value must be smaller than UpStreamBandwidth .	512
UpStreamBandwidth2	Upstream bandwidth limit in kbps when failing over to the secondary WAN (LTE) interface.	2048
RestrictedBandwidth2	The bandwidth limit in kbps allocated to the Restricted Class when failing over to the secondary WAN (LTE) interface. The Restricted Class has the higher priority than any other priority classes. This value must be smaller than UpStreamBandwidth2 .	512
Priority Class Bandwidth Allocation (DeviceInfo.Qos.BandwidthControl.Queue.)		
High	The guaranteed uplink bandwidth allocation weight for High Priority class traffic. It must be a value between 1 and 10.	5
Medium	The guaranteed uplink bandwidth allocation weight for Medium Priority class traffic. It must be a value between 1 and 10.	3

QoS Parameter Guide

Parameter	Description	Default Setting
Low	The guaranteed uplink bandwidth allocation weight for Low Priority class traffic. It must be a value between 1 and 10.	2
<i>DSCP to Priority Class Mapping (DeviceInfo.Qos.BandwidthControl.DSCPMapping.)</i>		
N (N = 0 – 63)	The priority class to be assigned to the packet that has the DSCP code equal to N in the IP header. Choose from: - Restricted - High - Medium - Low Note: Restricted class has the highest priority.	For N = 8, 10, 12, or 14, the default is Low. For N = 48 or 56, the default is High. For all other DSCP codes, the default is Medium.

Auto Provisioning Parameter Guide

Parameter	Description	Default Setting
<i>Auto Firmware Update (X_DeviceManagement.FirmwareUpdate.)</i>		
Method	Current operational method of auto firmware updating. Choose from: - Disabled: don't check for f/w upgrade from FirmwareURL. - System Start: Check for f/w upgrade from FirmwareURL just once on system start. - Periodically: Check for f/w upgrade from FirmwareURL on system start, and then periodically at the interval specified in the Interval parameter. - Time of Day: Check once at the given TimeOfDay value. Note: The first firmware upgrade check on system start is performed after a random delay of 0 to 30 seconds.	Disabled
TimeOfDay	Time of the day in hh:mm[+rr] format, valid when method is set to Time of Day.	00:00+30
Interval	When Method is set to Periodically, this is the number of seconds between each checking of f/w upgrade check from FirmwareURL . If value is 0, the device checks once only on system start (equivalent to setting Method to System Start).	0
RandomDelayRange	The range of delay in seconds inserted before the first attempt only. The minimum value is 0.	30
Suspend	Suspend firmware update until canceled.	false
FirmwareURL	URL of firmware package. URL must include scheme. Supported schemes are http:// and tftp://	

Auto Provisioning Parameter Guide

Parameter	Description	Default Setting
DnsLookupType	Controls what type of DNS record to lookup. Choose from: - A Record Only - SRV Record Only - Try Both	A Record Only
DnsSrvPrefix	Controls whether to add a standard prefix to the domain name when looking up a SRV Record. For HTTP and HTTPS, the prefix to add is <code>_http._tcp.</code> . For TFTP, the prefix to add is <code>_tfto._udp.</code> Choose from: - No Prefix - With Prefix - Try Both	No Prefix
Username	Username for authentication, if needed, if scheme is <code>http://</code>	
Password	Password for authentication, if needed, if scheme is <code>http://</code>	
ITSP Provisioning (X_DeviceManagement.ITSPProvisioning.)		
Method	Current operational method of Provisioning. Choose from: - Disabled: don't download from ConfigURL. - System Start: Download from ConfigURL just once on system start. - Periodically: Download from ConfigURL on system start, and then periodically at the interval specified in the Interval parameter. - Time of Day: Check once at the given TimeOfDay value. Note: First download on system start is performed after a random delay of 30 to 90 seconds if there is a firmware update scheduled at the beginning, or a random delay of 10 to 70 seconds.	System Start
Interval	When Method is set to Periodically , this is the number of seconds between download from ConfigURL . If value is 0, device downloads once only on system start (equivalent to setting Method to System Start).	0
TimeOfDay	Time of the day in <code>hh:mm[+rr]</code> format, valid when method is set to Time of Day .	00:00+30
ConfigURL	URL of config file.	<code>tftp://\$DHCPOPT66/\$MAC.xml</code>
DnsLookupType	Controls what type of DNS record to lookup. Choose from: - A Record Only - SRV Record Only - Try Both	A Record Only

Auto Provisioning Parameter Guide

Parameter	Description	Default Setting
DnsSrvPrefix	Controls whether to add a standard prefix to the domain name when looking up a SRV Record. For HTTP and HTTPS, the prefix to add is <code>_http._tcp..</code> For TFTP, the prefix to add is <code>_tfto._udp..</code> Choose from: - No Prefix - With Prefix - Try Both	No Prefix
Override	Defines which local settings can be overridden by this provisioning. Choose from: - All - All except user settings	All
GPRM0 to GPRM7	Non-volatile generic parameters that can be referenced in other parameters, such as ConfigURL .	
TPRM0 to TPRM3	Temporary variables used in scripts for ConfigURL . Please see the Polycom® OBi Edition Deployment Guide for examples on how to use these variables.	
OBiTALK Provisioning (X_DeviceManagement.Provisioning.)		
Method	Current operational method of Provisioning. Choose from: - Disabled: Don't download from ConfigURL. - System Start: Download from ConfigURL just once on system start. - Periodically: Download from ConfigURL on system start, and then periodically at the interval specified in the Interval parameter. - Time of Day: Check once at the given TimeOfDay value. Note: First download on system start is performed after a random delay of 30 to 90 seconds if there is a firmware update scheduled at the beginning, or a random delay of 10 to 70 seconds.	System Start
Interval	When Method is set to Periodically , this is the number of seconds between download from ConfigURL . If value is 0, device downloads once only on system start (equivalent to setting Method to System Start).	0
TimeOfDay	Time of the day in hh:mm[+rr] format, valid when method is set to Time of Day .	00:00+30
ConfigURL	URL of config file.	
DnsLookupType	Controls what type of DNS record to lookup. Choose from: - A Record Only - SRV Record Only - Try Both	A Record Only

Auto Provisioning Parameter Guide

Parameter	Description	Default Setting
DnsSrvPrefix	Controls whether to add a standard prefix to the domain name when looking up an SRV Record. For HTTP and HTTPS, the prefix to add is <code>_http._tcp..</code> For TFTP, the prefix to add is <code>_tfto._udp..</code> Choose from: - No Prefix - With Prefix - Try Both	No Prefix
GPRM0 to GPRM7	Non-volatile generic parameters that can be referenced in other parameters, such as ConfigURL .	
TPRM0 to TPRM3	Temporary variables used in scripts for ConfigURL . Please see the Polycom® OBi Edition Deployment Guide for examples on how to use these variables.	
User-Defined Macro 0–3 (\$UDM0 – \$UDM3)		
Value	The value can be any plain text or a valid canonical parameter name preceded by a \$ sign. For example: \$X_DeviceManagement.WebServer.Port Note: Here you MUST NOT enclose the parameter name following the \$ sign with braces or parentheses.	
ExpandIn	This is a comma-separated list of canonical parameter names, where the macro expansion can be used. As many as three parameter names can be specified. Specify ANY to allow the macro to expand in any parameter. Example: <code>X_DeviceManagement.HTTPClient.UserAgent</code> Note: There is no \$ sign in front of the parameter name. The macro can't be used in any parameter value if this value is set to blank (the default).	
SyntaxCheckResult	This is read only status value regarding the syntax of the UDM. Pass means that this UDM is valid. Otherwise, it shows the syntax error detected by the device either in the Value or ExpandIn parameters of the UDM.	

\$MACRO Expansion Supported by the Device

\$MACRO Expansion Supported by the Device

Macro Name	Description	Where It Can Be Used
MAC	Device MAC address, such as 9CADEF000000	ANY
MACC	Device MAC address with colons, such as 9C:AD:EF:00:00:00	ANY
mac	Device MAC address in lower case with colons, such as 9c:ad:ef:00:00:00	ANY
FWV	Firmware version, such as 1.0.3.1626	ANY
HWV	Hardware version, such as 2.8	ANY
IPA	Current device IP address, such as 192.168.15.100	ANY
DM	Device Model Name, such as OBi508	ANY
DMN	Device model number, such as 508	ANY
OBN	Device OBi number, such as 200123456	ANY
DSN	Device S/N, such as 88B01NA00000	ANY
GPRM _n <i>n</i> =0–7	Value of Auto Provisioning::GPRM_n	Auto Provisioning::ConfigURL , Auto Firmware Update::FirmwareURL
TPRM _n <i>n</i> =0-3	Value of Auto Provisioning::TPRM_n	Auto Provisioning::ConfigURL , Auto Firmware Update::FirmwareURL
UDM _n , <i>n</i> =0-3	Value of User-Defined Macro <i>n</i>::Value	The value of User-Defined Macro <i>n</i>::ExpandIn

\$MACRO Expansion Supported by the Device

Macro Name	Description	Where It Can Be Used
MAC	Device MAC address, such as 9CADEF000000	ANY
MACC	Device MAC address with colons, such as 9C:AD:EF:00:00:00	ANY
mac	Device MAC address in lower case with colons, such as 9c:ad:ef:00:00:00	ANY
FWV	Firmware version, such as 1.0.3.1626	ANY
HWV	Hardware version, such as 2.8	ANY
IPA	Current device IP address, such as 192.168.15.100	ANY

\$MACRO Expansion Supported by the Device

Macro Name	Description	Where It Can Be Used
DM	Device Model Name, such as OBi300 and OBi500	ANY
DMN	Device model number, such as 508	ANY
OBN	Device OBi number, such as 200123456	ANY
DSN	Device S/N, such as 88B01NA00000	ANY
GPRMn n=0–7	Value of Auto Provisioning::GPRMn	Auto Provisioning::ConfigURL , Auto Firmware Update::FirmwareURL
TPRMn n=0-3	Value of Auto Provisioning::TPRMn	Auto Provisioning::ConfigURL , Auto Firmware Update::FirmwareURL
UDMn, n=0-3	Value of User-Defined Macro n::Value	The value of User-Defined Macro n::ExpandIn

Device Admin Settings

The Device Admin web page includes the following configuration parameters.

Device Administration Parameter Guide

Parameter	Description	Default Setting
Web Server (X_DeviceManagement.WebServer.)		
Port	Web server port number.	80
AdminPassword	Administrator password, case-sensitive.	admin
UserPassword	User password, case-sensitive.	user
IVR (X_DeviceManagement.IVR.)		
Enable	Enables IVR for local configuration.	Yes
Password	IVR access password (must be all digits).	
Syslog (X_DeviceManagement.Syslog.)		
Server	IP address of the Syslog server where the device sends syslog debug messages to. If the value is blank, syslog is disabled.	
Port	Syslog server port number.	514
Level	Syslog message level.	7
TAG	A string of text no longer than 32 characters to prepend every syslog message sent out by this unit.	
HTTP Client (X_DeviceManagement.HTTPClient.)		

Device Administration Parameter Guide

Parameter	Description	Default Setting
UserAgent	Value of the User-Agent header in all HTTP Requests that are used in firmware upgrade and auto provisioning.	\$DM
TimeOut	A time limit specified in number of seconds such that any file download (firmware or configuration file) by the device via HTTP must be completed within this limit or the device aborts and concludes that the operation has failed for the reason of "taking too long to complete".	600
ProxyServer	Host name or IP address of the HTTP proxy server.	
ProxyServerPort	Destination port to connect to the HTTP proxy server. Range = [0:65535]. Don't choose a port at random.	80
ProxyAuthUsername	User name for proxy authentication.	
ProxyAuthPassword	Password for proxy authentication.	
BypassProxyServerForLocalAddresses	Enables BypassProxyForSubnets parameter.	No
BypassProxyForSubnets	List of intranet subnets, which bypass the proxy server.	
External Port Security (X_DeviceManagement.X_PortSecurity.)		
PCPort	Locks the PC port so that the handset does not allow any network traffic in and out of that port.	No
Remote PCAP Server (X_DeviceManagement.X_RPCAPD.)		
Enable	Enables PCAP (Packet Capture) server function on the handset.	No
Ports	PCAP server port number.	2002
Client	List of clients allowed to connect to this server. An empty list means everyone is allowed.	None
Packet Capture (X_DeviceManagement.X_RemotePCAP.)		
Enable	Enable Remote PCAP server and start the daemon according to the following parameters.	Disabled
Status	Running status of utility capture.	
Interface		Primary
Storage	Where the captured packets are stored. Internal Storage: Volatile memory space is no more than 10MB.	Internal Storage
RestartCaptureOnReboot	Automatically start the capture according to the current configuration as soon as the specified network interface is created.	Disable

Device Administration Parameter Guide

Parameter	Description	Default Setting
PromiscuousMode	The Promiscuous Mode.	Enable
WebAccessExcluded	Exclude traffic packets to or from the local web server.	Enable
PostponeFirmwareUpdate	Postpone auto firmware update if capturing is ongoing.	Enable
Platform CA n ($X_DeviceManagement.PlatformCACert.n.$), $n = 1$ or 2		
DownloadURL	URL to download certificate	None
MD5Checksum	MD5 checksum of the certificate file to be downloaded. Failure to provide this causes the device to try to download the same file on every reboot or restart.	None
CommonName	The common name set in the installed certificate. Read-only status field.	
FingerPrint	SHA1 fingerprint of the installed certificates.	
Obsolete	When set to true, the certificate is deleted from the device. Also, the certificate downloading process is ignored.	False
Custom Device Certificate n ($X_DeviceManagement.CustomDeviceCert.n.$), $n = 1$ or 2		
DownloadURL	URL to download certificate.	None
MD5Checksum	MD5 checksum of the downloaded certificate.	None
CommonName	The common name set in the installed certificate. Read-only status field.	None
FingerPrint	SHA1 fingerprint of the installed certificates.	None
Obsolete	When set to true, the certificate is deleted from the device. Also, the certificate downloading process is ignored.	False
TLSPlatform Profile n ($X_DeviceManagement.TLSPlatform.n.$), $n = 1$ or 2		
CipherSuite	The cipher suite to use in a TLS profile (the encryption algorithms to support in establishing a TLS connection according to the TLS profile specification configured on the handset).	None

Device Administration Parameter Guide

Parameter	Description	Default Setting
CACertList	The CA Certificate List to use in a TLS profile. Choice of: • Default • Default+P1 • Default+P2 • All • Platform1 • Platform2 • Platform1+2	Default
DeviceCert	The Device Certificate List to use in a TLS profile. Choice of: • Polycom • Custom1 • Custom2	Polycom

Device Update

The Device Update web page provides the following functions:

- [Firmware Update](#)
- [Backup Configuration](#)
- [Restore Configuration](#)
- [Reset Configuration](#)

Firmware Update

You can update the firmware for your handset from the native web page. The firmware file must be stored locally on a computer that you can access with a web browser.

To update the firmware

- 1 Select the **System Management – Device Update** menu on the side panel of the web page.
- 2 Click the **Browse** button in the **Firmware Update** section of the page. In a file browser window, select the firmware file.
- 3 Click the **Update** button to start the upgrade process.

The process takes about 30 seconds to complete.



Don't disconnect the power from the device during this procedure. If the new firmware is upgraded successfully, the device reboots automatically to start running the new firmware. Otherwise, the web page shows an error message explaining why the upgrade failed.

Possible Error Messages on Firmware Update Failure

The following table lists the possible error messages encountered when a firmware upgrade fails.

Error Messages for Firmware Update Failure

Error Message	Description	Suggested Solution
Firmware Package Checksum Error	A corrupted firmware package file was used for the update.	Check the file and / or redownload the firmware package and try again.
System Is Busy	The device is busy because one of the services in an active call or device provisioning is in progress.	Try to update again later.
Firmware Is Not Modified	The device is already running the same firmware as the one selected for update.	No need to upgrade.

Backup Configuration

The current configuration of the handset can be backed up and stored as a file in XML format at a user specified location. The default name of the file is `backupxxxxxxxxxxxx.xml`, where `xxxxxxxxxxxx` represents the MAC address of your handset.

When backing up a device's configuration, you can select one the following options before clicking **Backup**.

Backup Options

Option	Description	Default Setting
Incl. Running Status	If checked, the values of all status parameters are included in backup file. Otherwise, status parameters are excluded from the backup.	No
Incl. Default Value	If checked, the default values of parameters are included in the backup file. Otherwise, default values are excluded from the backup.	No
Use OBi Version	If not checked, the backup file uses XML tags that are compliant with the TR-104 standard. Otherwise, the backup file is stored in an OBi proprietary format where the XML tags aren't compliant with TR-104, but the file size is smaller and the file is more readable.	No

When the file browser window opens, you can change the filename and choose the location to save the backup file.



Different web browsers may handle this differently. If the operation is blocked due to the security setting of the web browser, you should change the security setting temporarily to allow this operation to complete.

Restore Configuration

When restoring the configuration to a previous backup copy, you need to specify the backup file you want to restore to by clicking **Browse** in the **Restore Configuration** section of the web page. Then, select the **Restore** button to start the process. The handset reboots automatically after the restoration is complete.



All passwords and PINs are excluded from the backup file. Hence, they aren't available to restore. Call history is excluded from the backup, but can be saved as an XML formatted file separately from the Call History web page.

Reset Configuration

The **Reset Configuration** function resets the handset to its factory default condition. Call history and various statistical information is removed at the same time. Resetting the device configuration should be used with extreme caution as the operation cannot be undone.

Service Providers Parameters

The Service Providers web pages show parameters for the provisioned service providers on your device. They include these sets of pages for each of the eight ITSP Profiles A through H.

- [General Settings](#)
- [SIP Settings](#)
- [RTP Settings](#)

ITSP Profile X (X = A, B, C, D, E, F, G, H)

ITSP profiles represent profiles for the service providers. Voice service profiles, described in the next set of web pages, represent the profiles that bind your device to the service providers.

General Settings

The following configuration parameters are available on this page.

ITSP Profile X – General Web Page (X = A, B, C, D, E, F, G, H) Parameter Guide

Parameter	Description	Default Setting
<i>ITSP Profile X – General (VoiceService.1.VoiceProfile.n.) for X = A, B, ..., H corresponding to n = 1, 2, ..., 8, respectively</i>		
Name	Human-readable string to identify the profile instance. Maximum length is 127 characters.	
SignalingProtocol	Signaling protocols for this ITSP.	SIP

ITSP Profile X – General Web Page (X = A, B, C, D, E, F, G, H) Parameter Guide

Parameter	Description	Default Setting
DTMFMethod	Method to pass DTMF digits to peer device. Choose from: Inband: DTMF tones are sent as inband audio signal RFC2833: DTMF tone events are relayed per RFC2833 SIPInfo: DTMF tones are relayed with SIP INFO request Auto: Method to use based on call setup negotiation (either Inband or RFC2833 can be negotiated).	Auto
InbandDTMFVolume	DTMF tone volume when sending inband DTMF	15
X_UseFixedDurationRFC2833DTMF	When relaying DTMF digit events on this trunk using RFC2833, the RFC2833 RTP packets normally keep streaming for as long as the digit is pressed. With this option set to TRUE, the device sends only one RTP digit event packet with a fixed duration of 150 ms regardless how long the digit has been pressed.	False
X_FixedDurationRFC2833DTMF	The fixed duration (in ms) to use when X_UseFixedDurationRFC2833DTMF is set to True.	16
DigitMap	A digit map to restrict the numbers that can be dialed or called with this service. Maximum length is 511 characters.	(1xxxxxxxxxx <1>[2-9]xxxxxxxx 011xx. xx.)
STUNEnable	Enables device to send a STUN binding request for its RTP port prior to every call.	No
STUNServer	IP address of domain name of the STUN Server to use.	
X_STUNServerPort	UDP listen port of the STUN Server.	3478
X_ICEEnable	Enables device to use ICE algorithm to find the best peer RTP address to forward RTP traffic for every call.	No
X_SymmetricRTPEnable	Enables device to apply symmetric RTP behavior on every call: That is, send RTP to peer at the address where incoming RTP packets are received from.	No

Service Provider Info (VoiceService.1.VoiceProfile.n.ServiceProviderInfo.) for X = A, B, ..., H corresponding to n = 1, 2, ..., 8, respectively)

Name	Human-readable string identifying this service provider. Maximum length is 127 characters.	
URL	Website of this service provider. Maximum length is 127 characters.	

ITSP Profile X – General Web Page (X = A, B, C, D, E, F, G, H) Parameter Guide

Parameter	Description	Default Setting
ContactPhoneNumber	Phone number to contact this service provider. Maximum length is 31 characters.	
EmailAddress	Email address to contact this service provider. Maximum length is 127 characters.	

SIP Settings

The following configuration parameters are available on this page.

ITSP Profile X – SIP Web Page (X = A, B, C, D, E, F, G, H) Settings Parameter Guide

Parameter	Description	Default Setting
<i>ITSP Profile X – SIP (VoiceService.1.VoiceProfile.n.SIP) for X = A, B, ..., H corresponding to n = 1, 2, ..., 8, respectively)</i>		
ProxyServer	Host name or IP address of the SIP proxy server.	
ProxyServerPort	Destination port to connect to the SIP server.	5060
ProxyServerTransport	Transport protocol to connect to SIP server. Choose from: • UDP • TCP • TLS	UDP
RegistrarServer	Host name or IP address of the SIP registrar. If a value is specified, device sends REGISTER to the given server. Otherwise, REGISTER is sent to ProxyServer .	
RegistrarServerPort	Destination port to connect to SIP registrar.	5060
X_XsiServer	Host name or IP address of the Broadsoft XSI application server.	
X_XsiServerPort	Listening port of the Broadsoft XSI application server. If not specified or 0, the default ports are 80 for HTTP and 443 for HTTPS.	0
X_XsiServerScheme	Scheme to access the Broadsoft XSI application server.	HTTP

ITSP Profile X – SIP Web Page (X = A, B, C, D, E, F, G, H) Settings Parameter Guide

Parameter	Description	Default Setting
UserAgentDomain	CPE domain string. If empty, device uses ProxyServer as its own domain to form its AOR (Address Of Record) or Public Address when constructing SIP messages (for example, in the FROM header of outbound SIP Requests). Note: If SPn Service::URI is specified, additional rules applied in forming the AOR. See the description of the URI parameter for more details and examples.	
OutboundProxy	Host name or IP address of the outbound proxy. Outbound proxying is disabled if this parameter is blank.	
OutboundProxyPort	Destination port to be used in connecting to the outbound proxy.	5060
X_OutboundProxyTransport	Controls the SIP transport for the outbound proxy server, which can be different from that of the proxy server. Choose from: • UDP • TCP • TLS • Follow ProxyServerTransport	Follow ProxyServerTransport
X_UserAgentContactFollowProxyServerTransport	If enabled, the user agent uses a Contact and Via transport that agrees with ProxyServerTransport .	No
X_BypassOutboundProxyInCall	Enables bypassing the OutboundProxy inside the SIP dialog.	No
RegistrationPeriod	Nominal interval between device register in seconds.	60
X_RegistrationMargin	Number of seconds before current registration expires that the device should re-Register (for example, 5 seconds). If value is less than one, it is interpreted as a fraction of the current expires value (for example, 0.1 of 60 seconds is 6 seconds). If value is 0 or blank, the device determines a proper margin on its own.	
TimerT1	Value of SIP timer T1 in ms.	500
TimerT2	Value of SIP timer T2 in ms.	4000
TimerT4	Value of SIP timer T4 in ms.	5000
TimerA	Value of SIP timer A in ms.	500
TimerB	Value of SIP timer B in ms.	32000
TimerD	Value of SIP timer D in ms.	32000

ITSP Profile X – SIP Web Page (X = A, B, C, D, E, F, G, H) Settings Parameter Guide

Parameter	Description	Default Setting
TimerE	Value of SIP timer E in ms.	500
TimerF	Value of SIP timer F in ms.	32000
TimerG	Value of SIP timer G in ms.	500
TimerH	Value of SIP timer H in ms.	32000
TimerI	Value of SIP timer I in ms.	5000
TimerJ	Value of SIP timer J in ms.	32000
TimerK	Value of SIP timer K in ms.	5000
InviteExpires	Invite request Expires header value in seconds.	60
ReInviteExpires	Reinvite Expires header value in seconds.	10
RegisterExpires	Register Expires header value in seconds (not used at the moment).	3600
RegistersMinExpires	Register Min-Expires header value in seconds (not used at the moment).	15
RegisterRetryInterval	Register retry interval in seconds.	30
X_RegisterRetryResponseCode	A set of SIP register error response codes and the corresponding retry delay (in seconds) specified in a digit map format. See the default value on the right as an example, where the value to the left of the colon of each rule represents a set of 3-digit response codes and the value to the right of the colon is the waiting time in seconds. If the waiting time is given as a range (with a '-'), a randomized waiting time within the specified range is used.	(<40[17]:w120> <40[34]:w120> <99[01]:w120-200> [4-9]xx)
DSCPMark	Diffserv code outgoing SIP packets.	0
X_SpoofCallerID	Allows outbound Caller ID spoofing. If set to Yes, device attempts to set the caller-id name and user id field in the FROM header to that of a remote caller in the case of a bridged call (from another trunk, such as another SP Service). Otherwise, device always its own account information to form the FROM header. Note that most service providers won't allow originating a call if the FROM header field doesn't match the account credentials. Enable this option only if you're sure that the service provider allows it.	No
X_UseRefer	Enables using SIP REFER for call transfer. If disabled, device bridges the call instead when performing a call transfer (which consumes some resources on the device).	Yes

ITSP Profile X – SIP Web Page (X = A, B, C, D, E, F, G, H) Settings Parameter Guide

Parameter	Description	Default Setting
X_ReferAOR	Enables using the target's AOR (Address of Record or public address) in Refer-To header of SIP REFER. If disabled, the target's Contact is used instead.	Yes
X_HoldReferee	Holds the Referee before a blind transfer if the call isn't placed on hold. This may allow reconnecting with the Referee if the blind call transfer fails.	No
X_Use302ToCallForward	Enables using the 302 response to INVITE for call forward. If disabled, device bridges the call legs instead when forwarding a call (and consumes some resources on the device).	Yes
X_UserAgentName	If a value is specified, device includes a User-Agent header in all SIP Requests, or a Server header in all SIP responses, that contains exactly the given value.	OBIHAI/\${DM}-\${FWV}
X_ProcessDateHeader	Enables the device to decode the DATE header sent by the ITSP in a 200 response to its REGISTER. The DATE header specifies the current GMT time and the device can use to adjust its local time and date without relying on NTP.	Yes
X_InsertRemotePartyID	Enables the device to include a Remote-Party-ID header in its outbound SIP INVITE to indicate to the ITSP the caller's preferred privacy setting (either full or none).	Yes
X_EnforcePAssertedIdentity	Take caller-ID from P-Asserted identity header only.	No
X_InsertPPreferredIdentity	Insert P-Preferred-Identity header in all outbound INVITE.	No
X_InsertPrivacyHdr	Inserts a 'Privacy:id' header in outbound INVITE for anonymous calls.	No
X_UseAnonymousFROM	Enables using "sip:anonymous@localhost" in FROM header of SIP INVITE when attempting to make an anonymous call.	No
X_SessionRefresh	Enables session refresh signaling (with SIP Re-INVITE) during a connected call. This allows the device to detect if the connection with the peer is broken abnormally so it can release the call. Disable this option if the ITSP doesn't support Re-INVITE sent from the client device.	Yes
X_SessionTimer	Enable standard session timer behavior based on RFC4028	No
X_SessionExpires	Session Expires before value. If session refresh is enabled, the device refreshes half-time before the session expires.	20

ITSP Profile X – SIP Web Page (X = A, B, C, D, E, F, G, H) Settings Parameter Guide

Parameter	Description	Default Setting
X_AccessList	A comma-separated list of IP addresses such that the device only accepts SIP requests coming from one of the given addresses. If the list is empty, the device accepts SIP requests from any IP address.	
X_InsertRTPStats	Enables the device to include a X-RTP-Stat header in a BYE request or 200 response to BYE request at the end of an established call. This header contains a summary of RTP statistics collected during the call.	Yes
X_MWISubscribe	Enables the device to SUBSCRIBE to the message-summary event package to support MWI and VMWI service. The device handles NOTIFY of this event package regardless of whether MWISubscribe is enabled.	No
X_MWISubscribeURI	Blank implies to use the same URL as REGISTER for the TO and FROM header as well as the Request-URI. Otherwise, if the URI doesn't contain '@', it's user as the user id field in TO/FROM header as well as the Request-URI, which are otherwise same as REGISTER. If the URI contains '@', it's used in the TO and FROM header as well as the Request-URI as is. The device forms the Request-URI of SUBSCRIBE the same way as the TO header, with an additional port number.	
X_MWISubscribeExpires	Periodic interval to renew SUBSCRIBE.	3600
X_ProxyServerRedundancy	Enables proxy redundancy feature on the device. To use this feature, device registration must be enabled and the SIP Registration Server or Outbound Proxy Server must be configured as a domain name.	No
X_SecondaryRegistration	Enables device to register with a secondary server in addition to the primary server. X_ProxyServerRedundancy must be enabled for this parameter to take effect.	No
X_CheckPrimaryFallbackInterval	Interval in seconds at which the device checks the primary fallback list of candidate servers.	60
X_CheckSecondaryFallbackInterval	Interval in seconds at which the device checks the secondary fallback list of candidate servers.	60

ITSP Profile X – SIP Web Page (X = A, B, C, D, E, F, G, H) Settings Parameter Guide

Parameter	Description	Default Setting
X_ProxyFailoverResponseCodes	A list of failure response codes specified in the form of a digit map string to trigger proxy failover. If only one-digit map is specified, it applies to REGISTER and INVITE requests. If two-digit maps are provided (separated by a comma), the first one applies to REGISTER and the second to INVITE.	([5-9] xx)
X_InviteFailoverWaitRegTime	Maximum time (in milliseconds) to wait for successful register failover to retry INVITE after failure.	32000
X_ProxyRequire	If this parameter isn't blank, the device includes a Proxy-Require header stating the value of this parameter in all SIP requests sent to the ITSP.	
X_MaxForward	Value for the Max-Forward header in all SIP requests sent by the device.	70
X_AcceptLanguage	If this parameter isn't blank, the device includes an Accept-Language header stating the value of this parameter in all SIP requests sent to the ITSP.	
X_DnsSrv	Enable DNS SRV lookup for the proxy server or the outbound proxy server.	Yes
X_DnsSrvAutoPrefix	Enables letting the device automatically prepend a standard prefix to the domain name when querying DNS Server to resolve the ProxyServer or OutboundProxy name as a SRV record. The standard prefix is <code>_sip._udp.</code> for SIP over UDP, <code>_sip._tcp.</code> for SIP over TCP, and <code>_sip._tls.</code> for SIP over TLS.	No
X_Support100rel	Enables support for RFC3262 (reliable provisional SIP responses). If enabled, the device announces this support in a SIP Supported header, and requires a caller to use this option if the caller also supports this feature.	No
X_UserEqPhone	Includes the parameter 'user=phone' in Request-URI and To-URI of outbound INVITE.	
X_UseTelURI	Enables using tel: in outbound SIP Request-URI and TO-URL	
X_CallWaitingIndication	Enables including an indication in an 18x response to the calling peer if this is a call- waiting situation. Choose from: - No - Alert-Info	No

ITSP Profile X – SIP Web Page (X = A, B, C, D, E, F, G, H) Settings Parameter Guide

Parameter	Description	Default Setting
X_DiscoverPublicAddress	Enables letting the device use the public IP address and port it has discovered as its SIP Contact address.	Yes
X_UsePublicAddressInVia	Enables using the discovered external IP address (instead of the unit's assigned local IP address) in outbound Via header.	No
X_PublicIPAddress	A static public IPv4 address, if specified, is used by the device to form its SIP Contact address.	
X_UseRport	Enables letting the device insert a blank rport parameter in the VIA header our outbound SIP messages. This option should be turned off if you're using port forwarding on the external router to route inbound SIP messages to the device.	Yes
X_DetectALG	Enables detecting upstream SIP ALG.	No
X_UseCompactHeader	Enables using compact form SIP message header names.	No
X_OmitContentLength	Omit Content-Length header if ProxyServerTransport and X_OutboundProxyTransport parameters are both UDP.	No
X_FaxPassThroughSignal	Selects the signaling method to indicate to the peer to switch to FAX passthrough. Choose from: - ReINVITE - RFC2833 - Auto - None	ReINVITE
X_IncludeMessageHash	Includes an MD5 hash of all the SIP headers in an XMD5-Hash header. A hash of the SDP is also included in an x-md5-hash SDP attribute.	No
X_EchoServer	Name or IP address of an echo server for SIP ALG detection.	
X_EchoServerPort	Listening of the echo server for SIP ALG detection.	5060
X_EnableRFC2543CallHold	Enables interpretation of call hold indication per RFC2543.	No
X_VerifyServerDomain	Enable verification of server domain against its certificate on a SSL/TLS connection.	No
X_RejectKeyResponseCode	SIP response code and phrase to inbound INVITE, when the user presses the End Call key.	

ITSP Profile X – SIP Web Page (X = A, B, C, D, E, F, G, H) Settings Parameter Guide

Parameter	Description	Default Setting
X_Sticky18x	Ignore further 18x responses without SDP upon reviving the first 18x with SDP to INVITE.	Yes
Feature Configuration (VoiceService.1.VoiceProfile.n.SIP), n = 1 - 8		
X_CallParkMethod	Select the method to use to park a call. Choice of: - Feature Code - REFER	Feature Code
X_DirectedCallPickupMethod	Select the method to use for directed call pickup. Choice of: - Feature Code - INVITE+Replaces	Feature Code
X_SharedLineMethod	Select the signaling method for shared line operation.	call-info
X_CallInfoSubscribeExpires	Call information subscription renewal interval in seconds. Set the value to 0 to disable subscription renewal.	3600
X_LineSeizeSubscribeExpires	Line-seize event subscription renewal interval in seconds.	15
Feature Codes (VoiceService.1.VoiceProfile.n.X_FeatureCode), n = 1 - 8		
DirectedCallPickup	Code to invoke Directed Call Pickup feature on the ITSP.	*97
CallPickup	Code to retrieve a parked call from the ITSP.	*88
Park	Code to park a call on the ITSP.	*68

RTP Settings

The following configuration parameters are available on this page.

ITSP Profile X – RTP Web Page (X = A, B, C, D, E, F, G, H) Parameter Guide

Parameter	Description	Default Setting
<i>ITSP Profile X – RTP (VoiceService.1.VoiceProfile.n.RTP.) for X = A, B, ..., H corresponding to n = 1, 2, ..., 8, respectively</i>		
LocalPortMin	Base of port range for tx/rx RTP with this SP.	16600 (X=A) 16800 (X=B) 17000 (X=C) 17200 (X=D) 17400 (X=E) 17600 (X=F) 17800 (X=G) 18000 (X=H)
LocalPortMax	Top of port range for tx/rx RTP with this SP.	16798 (X=A) 16998 (X=B) 17198 (X=C) 17398 (X=D) 17598 (X=E) 17798 (X=F) 17998 (X=G) 18198 (X=H)
KeepAliveInterval	Interval in seconds between sending keep alive packet on an RTP channel that is currently in idle (due to call hold for instance). RTP keepalive is disabled if the value of this parameter is set to 0.	0
DSCPMark	Diffserv code for outgoing RTP packets with this SP.	0
X_UseSSL	Enables forcing the device to send RTP over an SSL channel.	No
X_RefreshSession	Allow incoming RTP packets to refresh session.	Yes
X_SymmetricMedia	If incoming payload type changes unannounced, after 10 packets with the new payload type, decoding will switch to the new format. If symmetric media is enabled, outgoing packets will also be in the new format.	Yes
<i>ITSP Profile X – RTCP (VoiceService.1.VoiceProfile.n.RTP.RTCP.) for X = A, B, ..., H corresponding to n = 1, 2, ..., 8, respectively</i>		
Enable	Enables RTCP operation.	No
TxRepeatInterval	RTCP packet transmission interval in milliseconds.	10000
LocalCName	The canonical name to use in RTCP messages. If blank, the device uses <userid>@<local_IP_address> as its canonical name.	
X_RTCPMux	Enables using an rtcp-mux attribute in SDP (send and receive RTCP on the same port as RTP).	No

ITSP Profile *X* – RTP Web Page (*X* = A, B, C, D, E, F, G, H) Parameter Guide

Parameter	Description	Default Setting
X_VqPublishEnable	Enables VQ report sent to the proxy server using Publish method	false
X_VqPublishUrl	A Username or URL to send Voice Quality Report using Publish method	
X_VqPublishInterval	Interval in seconds between VQ reports; 0 or an empty value disables periodic reports	0
X_VqPublishOnSSRCChange	Enables VQ report when SSRC changes	true

ITSP Profile X - Jitter Buffer (VoiceService.1.VoiceProfile.n.RTP.JIB.) for X = A, B, ..., H corresponding to n=1, 2, ..., 8 respectively

Adaptive	Enable jitter buffer adaptation.	Yes
MaximumSize	Maximum jitter buffer size in milliseconds.	250
SetPoint	Initial playout delay in milliseconds.	60
Target	Target playout delay in milliseconds.	20
AdaptationSlope	Maximum adaptation slope in samples per 10 milliseconds.	16

Voice Services

The Voice Services parameters web pages show parameters for the voice services that are bound to the service providers on your device. They include these sets of pages.

- [SPn Service Settings \(n = 1, 2, 3, 4, 5, 6, 7, 8\)](#)
- [OBiTALK Service Settings](#)
- [Gateway Settings](#)
- [Page Group Settings](#)

SPn Service Settings (n = 1, 2, 3, 4, 5, 6, 7, 8)

The following configuration parameters are available on this page.

SPn Services (n = 1, 2, 3, 4, 5, 6, 7, 8) Settings

Parameter	Description	Default Setting
<i>SPn Service (VoiceService.1.VoiceProfile.1.Line.n.), n = 1 – 8</i>		
Enable	Enables this line.	True
X_ServProvProfile	Select a Service Provider profile for this service. Choices are A, B, C, D, E, F, G, or H.	A

SPn Services (n = 1, 2, 3, 4, 5, 6, 7, 8) Settings

Parameter	Description	Default Setting
<i>X_RingProfile</i>	Selects a Ring Profile to ring the handset for incoming calls on this service that are routed to it. The ringing pattern is taken from the given profile. Choices are A or B.	A
<i>X_CodecProfile</i>	Selects a Codec Profile for all calls on this service. Choices are A or B.	A
<i>X_InboundCallRoute</i>	Routing rule for directing incoming calls on this service. The default rule is to send all incoming calls to the handset. See the Call Routing chapter for a description of the syntaxes to specify this parameter.	DT1
<i>X_RegisterEnable</i>	Enables registration for this line. If set to True, the handset sends periodic SIP REGISTER to the service provider according to the settings in the ITSP Profile. Otherwise, the handset doesn't send any SIP REGISTER for the service.	True
<i>X_AcceptSipFromRegistrarOnly</i>	Accept SIP packets coming from the current registrar IP address only.	False
<i>X_NoRegNoCall</i>	Enables this option to disallow incoming and outgoing calls if registration with the service provider isn't successful.	False
<i>X_KeepAliveEnable</i>	Enables sending keep alive message. If set to True, the handset sends periodic keep-alive messages to the same server where a REGISTER request would be sent. The content of this message is the ASCII string keep-alive\r\n.	False
<i>X_KeepAliveExpires</i>	Keep alive period in seconds.	15
<i>X_KeepAliveMsgType</i>	The type of keep alive messages to send out periodically if keep-alive is enabled. Choice of: - keep-alive: The string keep-alive - empty: A blank line - stun: A standard STUN binding request. The handset uses the binding response to form its contact address for REGISTRATION - custom: use the value of X_CustomKeepAliveMsg - options: A SIP OPTIONS message - notify: A SIP NOTIFY message	keep-alive

SPn Services ($n = 1, 2, 3, 4, 5, 6, 7, 8$) Settings

Parameter	Description	Default Setting
<i>X_CustomKeepAliveMsg</i>	Defines the custom message to be used when X_KeepAliveMsgType is <i>custom</i>. The value has the following format: <pre>mtd=NOTIFY;event=<whatever>;user=<anyone></pre> Where <code>NOTIFY</code> can be replaced by any other SIP method, such as <code>PING</code> <code>event</code> is optional and is only applicable if method is <code>NOTIFY</code>. If event isn't specified, the 'keep-alive' event is used with <code>NOTIFY</code> <code>user</code> is optional. If not specified, the request-uri won't have a user ID, and the TO header field uses the same user ID as the FROM header, which is the local account user ID. If <code>user</code> is specified, it's used as the user ID in the Request-URI and TO header. SIP messages for keep-alive are sent only once without retransmission. Responses to the SIP messages are ignored by the handset.	None
<i>X_UserAgentPort</i>	UDP port where the handset sends and listens for SIP messages.	5060 ($n=1$) 5061 ($n=2$) 5062 ($n=3$) 5063 ($n=4$) 5064 ($n=5$) 5065 ($n=6$) 5066 ($n=7$) 5067 ($n=8$)
<i>X_UserAgentPorts</i>	A comma-separated list of as many as 10 alternative user agent ports to use when there is no response received from the SIP Registrar.	None
<i>DirectoryNumber</i>	Directory number associated with this service.	None
<i>X_DefaultRing</i>	Call waiting tone (as specified in the Ring Profiles) to play when there is a second incoming call.	1
<i>X_AcceptResync</i>	Control whether to accept a SIP NOTIFY request with <code>event=resync</code> to trigger a reboot of the handset so it can download new firmware or configuration upon boot up. Choice of: <code>no</code> (don't accept resync trigger) <code>yes with authentication</code> (accept after challenging the sender) <code>yes without authentication</code> (accept without challenging the sender)	yes without authentication

SPn Service — Debug Options (VoiceService.1.VoiceProfile.1.Line.n.), $n = 1 - 8$

SPn Services (n = 1, 2, 3, 4, 5, 6, 7, 8) Settings

Parameter	Description	Default Setting
X_SipDebugOption	Enables sending of SIP signaling debug information to the syslog server (if one is configured on the device). Choice of: <code>Disable</code> (do not send SIP signaling debug information) <code>Log All Messages</code> (log all messages) <code>Log All Except REGISTER Messages</code>	<code>Disable</code>
X_SipDebugExclusion	Comma-separated list of SIP methods to exclude from the debug log.	<code>None</code>

SPn Service — SIP Credentials (VoiceService.1.VoiceProfile.1.Line.n.SIP), n = 1 – 8

AuthUserName	The User ID to authenticate to a SIP UAS (User Agent Server) when an outbound SIP request sent by the handset is challenged by the UAS with a 401 or 407 response.	<code>None</code>
AuthPassword	The Password (corresponding to AuthUserName) to authenticate to a SIP UAS (User Agent Server) when an outbound SIP request sent by the handset is challenged by the UAS with a 401 or 407 response.	<code>None</code>
URI	This parameter affects the way the AOR is formed by the handset in outbound SIP Requests. The AOR has the format: <code>user@domain</code>. If the value of URI is empty, the handset gets the user portion of its AOR from the AuthUserName, and the domain portion the value of ITSP Profile's UserAgentDomain if it isn't empty, or that of the ProxyServer otherwise. If the value of URI isn't empty and doesn't contain "@", it is used as the user portion of the AOR while the domain portion is formed the usual way. If the value of URI contains @, it is interpreted as a full AOR and handset takes it as the AOR as is. Some examples: - Let ProxyServer = <code>sip.myitsp.com</code>, AuthUserName = <code>4089991123</code>, URI=<code>[empty]</code>, UserAgentDomain=<code>[empty]</code>, then AOR = <code>4089991123@sip.myitsp.com</code> - Change UserAgentDomain to <code>users.myitsp.com</code>, then AOR = <code>4089991123@users.myitsp.com</code> - Change URI to <code>bobdylan</code>, then AOR = <code>bobdylan@users.myitsp.com</code> - Change URI to <code>bobdylan@superusers.myitsp.com</code>, then AOR = <code>bobdylan@superusers.myitsp.com</code> Note: In all cases, the handset uses AuthUserName and AuthUserPassword to compute authorization if challenged by a 401 or 407 response.	<code>None</code>

SPn Services (n = 1, 2, 3, 4, 5, 6, 7, 8) Settings

Parameter	Description	Default Setting
X_XsiUserName	Username to authenticate a Broadsoft XSI application server. If not specified 'sip-userid@sip-proxy' is used.	Yes
X_XsiPassword	Password to authenticate a Broadsoft XSI application server. If not specified, the SIP password is used.	Yes
X_ContactUserID	An alternative user ID to be used in Contact header. Enter <i>Random</i> to let the handset generate a random one.	None
X_EnforceRequestUserID	Enforce incoming INVITE request user ID to match AuthUserName or X_ContactUserID .	False
SPn Service — Calling Features (VoiceService.1.VoiceProfile.1.Line.n.CallingFeatures.), n = 1 – 8		
CallerIDName	Display name to identify the subscriber. The display name field is usually inserted in a FROM header in outbound SIP requests (such as INVITE) for the purpose of displaying a Caller ID Name on the recipient's device.	None
MaxSessions	The maximum number of simultaneous calls that can be established on this service.	4
CallForwardUnconditionalEnable	Enables call forwarding of all calls unconditionally by the handset. If CallForwardUnconditionalNumber is blank, this parameter is treated as if it has been set to False. Note: You can set this parameter from the handset using a Star Code.	False
CallForwardUnconditionalNumber	Directory number to forward all incoming calls on this service unconditionally. Maximum Length is 127 characters. Note: You can set this parameter from the handset using a Star Code.	None
CallForwardOnBusyEnable	Enables call forwarding of all incoming calls when the handset is busy. If CallForwardOnBusyNumber is blank, this parameter is treated as if it has been set to False. The handset is considered busy if one of the following conditions holds: - This service already reaches the limit of simultaneous calls as specified in MaxSessions - DND (Do Not Disturb) Service is enabled on this service if the call is routed to the handset where the handset is in a busy state (such as ringing, dialing, playing reorder, or already having 2 calls in progress) Note: You can set this parameter from the handset using a Star Code.	False
CallForwardOnBusyNumber	Directory number to forward all incoming calls on this service when the handset is busy. Maximum Length is 127 characters. Note: You can set this parameter from the handset using a Star Code.	None

SPn Services (n = 1, 2, 3, 4, 5, 6, 7, 8) Settings

Parameter	Description	Default Setting
<i>CallForwardOnNoAnswerEnable</i>	Enables call forwarding of all incoming calls when the call isn't answered after a period as specified in CallForwardOnNoAnswerRingCount . If CallForwardOnNoAnswerNumber is blank, this parameter is treated as if it has been set to False. Note: You can set this parameter from the handset using a Star Code.	False
<i>CallForwardOnNoAnswerNumber</i>	Directory number to forward all incoming calls when the call isn't answered after a period specified in CallForwardOnNoAnswerRingCount . Note: You can set this parameter from the handset using a Star Code.	None
<i>CallForwardOnNoAnswerRingCount</i>	Number of rings to be considered by the handset as no answer to an incoming call. Note: 1 ring is approximately 6 seconds.	2
<i>X_BlockedCallers</i>	A comma-separated list of as many as 10 caller numbers to block from calling this service.	None
<i>X_MailboxID</i>	The mailbox ID to subscribe MWI with.	None
<i>X_CheckVoiceMailNumber</i>	The number to call to check voicemail.	None
<i>MWIEnableMask</i>	The set of handsets that are to receive the MWI or VMWI notifications.	
<i>X_VMWIEnableMase</i>	It is a bit mask. Each bit represents a handset. So 1023 (0x3ff) represents all 10 handsets. 4 (0x4) represents handset 3.	
<i>X_MWIRoute</i>	SIP/NOTIFY Routing Rules to enable MWI signals on MWI Notifications.	
<i>X_VMWIRoute</i>	SIP/NOTIFY Routing Rules to enable VMWI signals on MWI Notifications.	
<i>MessageWaiting</i>	This state parameter indicates if there are any new messages for this subscriber on the service provider's voicemail system.	No
<i>MessageCount</i>	Displays count of new messages, in format new/old (urgent – new/urgent – old).	None
<i>AnonymousCallBlockEnable</i>	Enables blocking Anonymous Calls on this service. Anonymous calls are rejected with a SIP 486 (Busy) response and Call Forward On Busy service isn't applied. Note: Users can set this parameter from the handset with a Star Code.	No

SPn Services (n = 1, 2, 3, 4, 5, 6, 7, 8) Settings

Parameter	Description	Default Setting
<i>AnonymousCallEnable</i>	Enables masking Caller-ID information for all outgoing calls. If enabled, the called party sees the call as coming from an anonymous caller. Note: Users can set this parameter from the handset with a Star Code.	No
<i>DoNotDisturbEnable</i>	Enables Do Not Disturb Service. If enabled, all incoming calls on this service are treated as if the device is busy. Note: Users can set this parameter from the handset with a Star Code.	No
<i>X_SRTP</i>	Enables SRTP. Choose one of: <code>Disable SRTP</code> = don't use SRTP for all calls. The call fails if the peer insists on using SRTP only. <code>Use SRTP Only</code> = Require all calls to use SRTP. The call fails if the peer doesn't support SRTP. <code>Use SRTP When Possible</code> = Use SRTP for a call if the peer supports SRTP. Otherwise, fall back to use regular unencrypted SRTP.	Disable SRTP
<i>X_ASFeatureEventSubscribe</i>	Enables subscription to the as-feature-event package.	False
<i>X_ConferenceBridge</i>	The number of an external conference bridge to use with calls on this service. Note: If the number also specifies the service to use, such as SP1(bridge@xyz-domain.com), the phone calls the number as it is on the given service. Otherwise, the phone applies its digit map and outbound call route setting to determine which service to use for the call.	cbridge

OBiTALK Service Settings

The following configuration parameters are available on this page.

OBiTALK Service Settings Parameter Guide

Parameter	Description	Default Setting
<i>OBiTALK Service Settings (VoiceService.1.X_P2P.1.)</i>		
Enable	Enables the OBiTALK Service (the built-in free voice service that comes with every OBi Device).	Yes
DisplayNumber	The number to display on the handset screen for a line or service.	
LocalPort	The UDP or TCP port used by the device to send and listens for OBiTALK messages.	10000

OBiTALK Service Settings Parameter Guide

Parameter	Description	Default Setting
TryMultiplePorts	Enables the device to try a few random UDP ports until it can successfully join the OBiTALK network.	No
Transport	Select the transport to connect the device to OBiTALK.	UDP/TCP
ServerAddress	OBiTALK server IP address (should not be empty in normal operation).	
LastRegistrarAddress	IP address and port number of the last registrar address used.	None
TLSListenerPort	OBiTALK TLS server listen port (443 in normal operation).	443
DisplayName	Display name to identify the subscriber, for the purpose of displaying a Caller ID Name on the recipient's device.	
DigitMap	Digit map to restrict numbers that can be dialed or called with this service.	(<ob>xxxxxxxxx x obxxxxxxxxx)
InboundCallRoute	Routing rule for directing incoming calls on this service. The default rule is to send all incoming calls to the handset(dt).	DT1
RingProfile	Selects a Ring Profile to ring the handset with when an incoming call is routed to the handset. Choose from A or B.	A
CodecProfile	Selects a Codec Profile to be used for all calls on this service. Choose from A or B.	A
DefaultRing	Call waiting tone (as specified in the Ring Profiles) to play when there is a second incoming call.	2
DTMFMethod	Method to pass DTMF digits to peer device. Choose from: Inband: DTMF tone are sent as inband audio signal RFC2833: DTMF tone events are relayed per RFC2833 SIPInfo: DTMF tones are relayed with SIP INFO request Auto: Method to use based on call setup negotiation (either Inband or RFC2833 can be negotiated)	Auto
FixedDurationRFC2833DTMF	The fixed duration (in groups of 10 milliseconds) to use when UsedFixedDurationRFC2833DTMF is True.	16
SymmetricMedia	If incoming payload type changes unannounced, after ten packets with the new payload type, decoding will switch to the new format. If symmetric media is enabled, outgoing packets also uses the new format.	Yes

OBiTALK Calling Features Parameter Guide

Parameter	Description	Default Setting
<i>OBiTALK Service – Calling Features (VoiceService.1.X_P2P.1.CallingFeatures.)</i>		
CallForwardUnconditionalEnable	Enables call forwarding of all calls unconditionally by the device. If CallForwardUnconditionalNumber is blank, this parameter is treated as if it has been set to <i>No</i> . Note: Users can set this parameter from the handset with a Star Code.	No
CallForwardUnconditionalNumber	Directory number to forward all incoming calls on this service unconditionally. Maximum length is 127 characters. Note: Users can set this parameter from the handset with a Star Code.	
CallForwardOnBusyEnable	Enables call forwarding of all incoming calls when the device is busy. If CallForwardOnBusyNumber is blank, this parameter is treated as if it has been set to <i>No</i> . Device is considered busy if one of the following conditions holds: - This service already reaches the limit of simultaneous calls as specified in MaxSessions - DND (Do Not Disturb) Service is enabled on this service. If the call is routed to the handset port where the handset is in a busy state (such as ringing, dialing, playing reorder, or already having 2 calls in progress) Note: Users can set this parameter from the handset with a Star Code.	No
CallForwardOnBusyNumber	Directory number to forward all incoming calls on this service when the device is busy. Maximum length is 127 characters. Note: Users can set this parameter from the handset with a Star Code.	
CallForwardOnNoAnswerEnable	Enables call forwarding of all incoming calls when the call isn't answered after a period as specified in CallForwardOnNoAnswerRingCount . If CallForwardOnNoAnswerNumber is blank, this parameter is treated as if it has been set to <i>No</i> . Note: Users can set this parameter from the handset with a Star Code.	No
CallForwardOnNoAnswerNumber	Directory number to forward all incoming calls when the call isn't answered after a period specified in CallForwardOnNoAnswerRingCount . Note: Users can set this parameter from the handset with a Star Code.	
CallForwardOnNoAnswerRingCount	Number of rings to be considered by the device as no answer to an incoming call. Note: 1 ring is approximately 6 seconds.	2

OBiTALK Calling Features Parameter Guide

Parameter	Description	Default Setting
BlockedCallers	A comma-separated list of as many as 10 caller numbers to block from calling this service.	
MaxSessions	The maximum number of simultaneous calls that can be established on this service.	2
AnonymousCallBlockEnable	Enables blocking Anonymous Calls on this service. Anonymous calls are rejected with a SIP 486 (Busy) response and Call Forward On Busy service isn't applied. Note: Users can set this parameter from the handset with a Star Code.	No
AnonymousCallEnable	Enables masking Caller-ID information for all outgoing calls. If enabled, the called party sees the call as coming from an anonymous caller. Note: Users can set this parameter from the handset with a Star Code.	No
DoNotDisturbEnable	Enables Do Not Disturb Service. If enabled, all incoming calls on this service are treated as if the device is busy. Note: Users can set this parameter from the handset with a Star Code.	No

Jitter Buffer (VoiceService.1.X_P2P.1.JIB.)

Adaptive	Enables jitter buffer adaptation.	True
MaximumSize	Maximum jitter buffer size in ms.	250
SetPoint	Initial play out delay in ms.	60
Target	Target play out delay in ms.	20
AdaptationSlope	Maximum adaptation slope in samples per 10 ms.	16

Gateway Settings

The following configuration parameters are available on this page.

Gateways Parameter Guide

Parameter	Description	Default Setting
<i>Voice Gateway n (VoiceService.1.X_VoiceGateway.n.) for $n = 1 - 8$</i>		
Enable	Enables this voice gateway.	Yes
Name	An arbitrary user-friendly name to identify this gateway (optional).	

Gateways Parameter Guide

Parameter	Description	Default Setting
AccessNumber	The gateway's OBiTALK number, including trunk information, such as: PP(ob200112334) or PP(ob200112334) If the value is blank, the device treats this VG as disabled. Starting with release 1.2, this can also be set to a SIP URL, such as: SP1(sip.mycompany.com:5060) or SP2(192.168.15.113)	
DigitMap	DigitMap for this VG. It can be referenced as (Mvgn).	(x.x)
AuthUserID	A user ID to authenticate with the gateway.	
AuthPassword	A password to authenticate with the gateway.	

Page Group Settings

The following configuration parameters are available on this page.

Page Groups Parameter Guide

Parameter	Description	Default Setting
Page Group 1 or 2 (VoiceService.1.X_PageGroup.n.), n = 1, 2		
GroupName	A user friendly name to label the group on the handset user interface.	None
MulticastAddress	Must be a valid IPv4 Multicast Address.	224.1.1.100
MulticastPort	Port to use for multicast.	65322
TTL	TTL value of outgoing (multicast) RTP packets.	2
ParticipantName	Name to identify this participant to the group.	None
AudioCodec	Audio codec to use for outgoing page.	G711U
TxPacketSize	RTP transmission packet size (in ms).	20
RTCPTxInterval	RTCP transmission interval (in ms) when talking.	0
SilenceSuppression	Enable silence suppression when talking.	False
PlayToneOnIncomingPage	Play a short paging tone on receiving an incoming page.	True
StartTalkingOnJoin	Start talking immediately when joining the group.	Ture
TalkingAlertTone	A short call waiting tone plays periodically to remind you the device is in talking mode.	CWT10
SwitchToTalkModeDigit	Digit to switch from listening mode to talking mode.	*
SwitchToListenModeDigit	Digit to switch from talking mode to listening mode.	#

DECT Wireless

The DECT Wireless web page shows parameters for the wireless handset. They include these sets of pages.

- [System Settings](#)
- [Registration Settings](#)
- [Handsetn \(n = 1, 2, ..., 9, 10\) Settings](#)

System Settings

The following configuration parameters are available on this page.

System Parameter Guide

Parameter	Description	Default Setting
System - DECT Base Information (VoiceService.1.X_HS.)		
TargetVersion	Firmware version.	
RFPI	Radio fixed part identity of the base.	
DectType	Displays the DECT device region.	
BaseName	Name of the base unit.	
TargetFW	Target firmware file to use. For debugging use only.	
TargetFWUpgrade	Enables target firmware upgrade.	Yes
TargetFWDowngradeAllowed	Allow the DECT target firmware to be downgraded when the full firmware bundle is an older version.	No
IntercomEnable	Enables handset intercom function.	Yes
Handset Status		
HandsetnStatus, n=1-10	Displays the handset status.	
Handset Information		
HandsetnType, n=1-10	Displays the handset type.	
HandsetnIPEI, n=1-10	Displays the handset's International Portable Equipment Identity (IPEI).	
Handset Firmware		
HandsetnFWVersion, n=1-10	Displays the handset firmware version.	
HandsetFW	Handset firmware file to use. For debugging use only.	
HandsetFWDowngradeAllowed	Allow handset firmware to be downgraded when the full firmware bundle is an older version.	No
HandsetnFWUpgrade, n=1-10	Enable handset firmware upgrade.	Yes

System Parameter Guide

Parameter	Description	Default Setting
Handset Locator		
FindHandsetAll	Locate all handsets. Equivalent to pressing the "FIND" button on the unit.	
FindHandsetn, n=1-10	Locate individual handset(s).	

Registration Settings

The following configuration parameters are available on this page.

Registration Parameter Guide

Parameter	Description	Default Setting
Registration - DECT Handset Registration (VoiceService.1.X_HS.)		
RegistrationWindow	Registration window status.	
OpenRegistration	Opens handset registration.	
RegisteredHandsets	Displays a list of registered handsets.	
DeleteHandsetAll	Deletes all registered handsets.	
DeleteHandsetn, n=1-10	Deletes the selected handset.	

Handsetn (n = 1, 2, ..., 9, 10) Settings

The following configuration parameters are available on this page.

Handsetn Settings Parameter Guide (n = 1, 2, ..., 9, 10)

Parameter	Description	Default Setting
Handsetn - Settings (VoiceService.1.X_HS.n.), n = 1, 2, ..., 9, 10		
Enable	Enables the handset.	Yes
Name	Sets the name of the handset.	

Handsetn Settings Parameter Guide (n = 1, 2, ..., 9, 10)

Parameter	Description	Default Setting
DigitMap	Restricts the numbers that can be dialed or called from the handset. If the caller dials a number that isn't allowed by the digit map, the device plays a SIT tone followed by a short error message to let the caller know that the dialed number is invalid.	<pre>([1-9]x?* (Mpli) [1-9]S9 [1-9] [0-9]S9 911 [67]XX * * 0 *** #S4 # [0-8] #9x **81 (Mbt) **82 (Mbt2) **1 (Msp1) **2 (Msp2) **3 (Msp3) **4 (Msp4) **9 (Mpp) (Mpl i))</pre>
OutboundCallRoute	After the caller dials a number that is acceptable according to the DigitMap , the device uses this outbound call routing rule to determine that service to make this call with. If no appropriate call route is found, the device plays a SIT tone followed by a short error message to let the caller know that there is no call route to place the call.	<pre>{ ([1-9]x?* (Mpli)) :pp}, {# 0 :ao}, {#1:dt1}, {#2:dt2}, { # 3:dt3}, {#4:dt4}, {#5:dt5} , {#6:dt6}, {#7:dt7}, {#8:dt 8 }, {#*:dt1,dt2,dt3,dt4,dt 5,dt6,dt7,dt8}, { (<6:park>XX<;s=1>) :pk}, { (<7:pickup>XX<;d=0>) :pk} , { (<**82:> (Mbt2)) :bt2}, { (< **81:> (Mbt)) :bt}, {**0:aa } , {***:aa2}, { (<**1:> (Msp1)) :sp1}, { (< * *2:> (Msp2)) :sp2}, { (<**3: > (Msp3)) :sp3}, { (<**4:> (Msp4)) :sp4}, { (< * *9:> (Mpp)) :pp}, { (Mpli) :p l i}</pre>
OutboundServices	List of services available for dialing out.	sp1, pp

Handsetn Settings Parameter Guide (n = 1, 2, ..., 9, 10)

Parameter	Description	Default Setting
CallReturnDigitMaps	Call Return is the service where you can call the last caller by dialing a star code (*69 by default). The device implements this service by remembering the number of the last caller in memory. However, the stored information doesn't include any dialing prefix to tell the device which voice service to use to call back the last caller. This list of digit maps serve the purpose of mapping a caller's number to one that includes the desired dialing prefix used exclusively for call return service.	{pli: (xx.)}, {sp1: (<*1>x x .)}, {sp2: (<*2>xx.)}, {sp3: (<*3>xx.)}, {sp4: (<*4>xx.)}, {bt1: (< * *81>xx.)}, {bt2: (<*82>xx .)}, {pp: (<*9>xx.)}
PrimaryLine	The "primary line" is the service that doesn't require any access code prefix (such as **1 or **9) when dialing. It is the default service to be used for making the call when no explicit access code prefix is entered. This parameter indicates to the device which voice service is considered as the primary line when dialing out from the handset. Choose from: • SP1 Service (code = sp1) • SP2 Service (code = sp2) • SP3 Service (code = sp3) • SP4 Service (code = sp4) • SP5 Service (code = sp5) • SP6 Service (code = sp6) • SP7 Service (code = sp7) • SP8 Service (code = sp8) • OBiTALK Service (code = pp1) The device process the parameter by substituting of the occurrences of pli and (Mpli) in DigitMap, OutboundCallRoute, and CallReturnDigitMaps with the corresponding code and (Mcode). For example, if PrimaryLine = sp3, then all occurrences of pli and (Mpli) are substituted internally with sp3and (Msp3).	SP1 Service

Handsetn - Calling Features (VoiceService.1.X_HS.n.CallingFeatures.), n = 1, 2, ..., 9, 10

Handsetn Settings Parameter Guide (n = 1, 2, ..., 9, 10)

Parameter	Description	Default Setting
CallerIDEnable	Enables Caller ID Signal generation. This option can be set to Yes even if the attached handset isn't capable of displaying Caller ID. There is no harm in sending Caller ID signal while the handset is in the on hook state.	Yes
CallWaitingCallerIDEnable	Enables Call Waiting Caller ID (CWCID) Signal generation. The CWCID signal is sent to the handset when it is in the off hook state. It starts with a handshake between the device and the attached handset, by exchanging audible short tones. The device proceeds with the transmission of the remaining Caller ID signal only if the handshake succeeds (with a handset is capable of displaying CWCID). In that case the handset mutes the handset earpiece until the CWCID signal is complete. Some users however may still find the audible handshake tones objectionable, especially if their handsets don't support CWCID. Set this option to No if you don't want the CWCID feature, or don't have handsets that can display CWCID.	Yes
MWIEnable	Enables MWI Signal (stutter dial tone) generation. If enabled, any SP voice service enabled on the device that has MWI Service enabled triggers the generation of stutter dial tone if there are new voicemails for the subscriber on the service provider's voicemail system.	Yes
VMWIEnable	Enables VMWI Signal generation. If enabled, any SP voice service enabled on the device that has VMWI Service enabled triggers the generation of VMWI signal if there are new voicemails for the subscriber on the service provider's voicemail system.	Yes

Handsetn Settings Parameter Guide (n = 1, 2, ..., 9, 10)

Parameter	Description	Default Setting
CallTransferEnable	Enables Call Transfer. If enabled, you initiate Call Transfer by hanging up the handset in one of the following scenarios: • One call on hold while a second outgoing call ringing (Case 1) • One call on hold while a second outgoing call connected (Case 2) • One call connected while a second outgoing call ringing (Case 3) • 3-way conference with both calls connected (Case 4) If Call Transfer is disabled, hanging up the handset in the above scenarios ends all the calls except for the one that is holding, which remains on hold (Cases 1 and 2).	Yes
ConferenceCallEnable	Enables 3-way Conference Call w/ local audio mixing. If enabled, you initiate Conference Call by hook flashing the handset in one of the following scenarios: • One call on hold while a second outgoing call ringing (Case 1) • One call on hold while a second outgoing call connected (Case 2) Case 1 is an early conference, where the second conferencee is still ringing. The other two parties may converse while hearing ringback tone in the back-ground until the third party answers. In either case, you can end the call with the second conferencee by hook flashing another time and the call reverts to a 2-way call. If Conference Call service is disabled, then hook flashing the handset resumes the holding call but ends the second outgoing call in Case 1, and swaps between the two calls in Case 2 (as in a call waiting situation).	Yes

Handsetn Settings Parameter Guide (n = 1, 2, ..., 9, 10)

Parameter	Description	Default Setting
UseExternalConferenceBridge	Enables using an external conference bridge for conference calls (SIP only). In addition, the following rule <code>{cbridge:SPx(bridge-userid)}</code> must also be added to the handset port's OutboundCallRoute parameter, where x=1,2,3,4, and bridge-userid the userid of the conference bridge SUA. Note that the keyword cbridge is hard-coded and must not be changed.	No
CallWaitingEnable	Enables call waiting service. Call Waiting is the situation where a new incoming call is routed to the handset port when there is already another call connected. If this service is enabled, the device plays the call-waiting tone to alert you, as well as generates the CWCID signal if CWCID is enabled. You can then swap between the two calls by hook flashing. If the service is disabled, the device rejects the incoming call as busy. Note: Users can set this parameter from the handset with a Star Code.	Yes
ToneProfile	Selects a Tone Profile for call progress tone generation. Choose from A or B.	A
StarCodeProfile	Selects a Star Code Profile for interpreting Star Codes you enter. Choose from: - None - A - B If set to None, no star code is recognized by the device.	A
LastDialedNumber	Last number dialed out on the handset.	
LastCallerNumber	Last caller's number that rings the handset.	
RepeatDialInterval	Interval in seconds between retry in a repeat dial operation.	30
RepeatDialExpires	Duration of time in seconds when a repeat dial operation remains active.	1800

Handsetn Settings Parameter Guide (n = 1, 2, ..., 9, 10)

Parameter	Description	Default Setting
MOHServiceNumber	The number to call to get music streamed to the remote party when the remote party is placed on hold.	
PlaySITOnCallFailureCodes	A list of (3-digit) error respons codes on outbound calls to trigger SIT with optional announcement of the error. The device plays fast busy tone without any announcement for all other call failure codes. The codes must be specified collectively as a digit map.	([4-9] xx)
PlaySITWithAnnouncement	Enables including announcement of the error when an outbound call fails.	Yes

Timers (VoiceService.1.X_HS.n.Timer.), n = 1, 2, ..., 9, 10

ReorderDelayTime	Delay (in ms) to start reorder tone after peer ends call.	5500
DigitMapLongTimer	Default number (in seconds) when the digit map processor should timeout waiting for more digits for matching patterns with an unspecified length.	10
DigitMapShortTimer	Default number (in seconds) when the digit map processor should timeout waiting for more digits when at least one pattern with specific length has matched.	2

Codec Profiles

The Codec Profiles parameters web pages include one page for each codec profile.

Codec Profile X Web Page (X = A, B)

The following configuration parameters are available on each page.

Codec Profile X Web Page (X = A, B)

Parameter	Description	Default Setting
Codec Profile X – G711U Codec (VoiceService.1.VoiceProfile.1.Line.n.Codec.List.1.) n = 1 or 2 corresponding to X = A or B, respectively		
Codec	Codec name.	PCMU
BitRate	Bit rate in bits/sec. Note: Informational only, not configurable.	64000

Codec Profile X Web Page (X = A, B)

Parameter	Description	Default Setting
Enable	Enables this codec.	Yes
SilenceSuppression	Enables silence suppression for this codec.	No
PacketizationPeriod	Packet size in ms.	20
Priority	Priority assigned to this codec (1 is the highest).	3
PayloadType	Standard payload type for this codec.	0

Codec Profile X – G711A Codec (VoiceService.1.VoiceProfile.1.Line.n.Codec.List.2.) n = 1 or 2 corresponding to X = A or B, respectively

Codec	Codec name.	PCMA
BitRate	Bit rate in bits/sec. Note: Informational only, not configurable.	64000
Enable	Enables this codec.	Yes
SilenceSuppression	Enables silence suppression for this codec.	No
PacketizationPeriod	Packet size in ms.	20
Priority	Priority assigned to this codec (1 is the highest).	4
PayloadType	Standard payload type for G711-alaw. Note: Informational only, not configurable.	8

Codec Profile X – G729 Codec (VoiceService.1.VoiceProfile.1.Line.n.Codec.List.3.) n = 1 or 2 corresponding to X = A or B, respectively

Codec	Codec name.	G729
BitRate	Bit rate in bits/sec. Note: Informational only, not configurable.	8000
Enable	Enables this codec.	Yes
SilenceSuppression	Enables silence suppression for this codec.	No
PacketizationPeriod	Packet size in ms.	20
Priority	Priority assigned to this codec (1 is the highest).	5
PayloadType	Standard payload type for G729. Note: Informational only, not configurable.	18

Codec Profile X – G726R32 Codec (VoiceService.1.VoiceProfile.1.Line.n.Codec.List.4.) n = 1 or 2 corresponding to X = A or B, respectively

Codec	Codec name.	G726-32
BitRate	Bit rate in bits/sec. Note: Informational only, not configurable.	32000

Codec Profile X Web Page (X = A, B)

Parameter	Description	Default Setting
Enable	Enables this codec.	Yes
SilenceSuppression	Enables silence suppression for this codec.	No
PacketizationPeriod	Packet size in ms.	20
Priority	Priority assigned to this codec (1 is the highest).	7
PayloadType	Dynamic Payload type for this codec. Valid range is 96–127.	104

Codec Profile X – iLBC Codec (VoiceService.1.VoiceProfile.1.Line.n.Codec.List.8.) n = 1 or 2 corresponding to X = A or B, respectively

Codec	Codec name.	iLbc
BitRate	Bit rate in bits/sec. Note: Informational only, not configurable.	13333
Enable	Enables this codec.	No
SilenceSuppression	Enables silence suppression for this codec.	No
PacketizationPeriod	Packet size in ms.	30
Priority	Priority assigned to this codec (1 is the highest).	6
PayloadType	Dynamic Payload type for this codec. Valid range is 96–127.	98

Codec Profile X – OPUS Codec (VoiceService.1.VoiceProfile.1.Line.n.Codec.List.10) n = 1 or 2 corresponding to X = A or B, respectively

Codec	Codec name.	OPUS
BitRate	Bit rate in bits/sec. Note: Informational only, not configurable.	20000
Enable	Enables this codec.	Yes
SilenceSuppression	Enables silence suppression for this codec.	No
PacketizationPeriod	Packet size in ms.	20
Priority	Priority assigned to this codec (1 is the highest).	2
PayloadType	Dynamic Payload type to be used to indicate this event.	109
UseInbandFEC	Enables use in band FEC when appropriate.	No

Codec Profile X – Telephone Event (VoiceService.1.VoiceProfile.1.Line.n.Codec.X_TelephoneEvent.) n = 1 or 2 corresponding to X = A or B, respectively

Codec	Codec Name for this RTP event, as used in SDP.	telephone-event
Enable	Enables this codec.	Yes

Codec Profile X Web Page (X = A, B)

Parameter	Description	Default Setting
PayloadType	Payload type to be used for RFC2833 telephone (DTMF) events. Valid range is 96–127.	101
Codec Profile X – Encap RTP (VoiceService.1.VoiceProfile.1.Line.n.Codec.X_EncapRTP.) n = 1 or 2 corresponding to X = A or B, respectively		
Codec	Codec Name. This codec is used to encapsulate RTP packets during a packet loopback call.	encaprtpt
PayloadType	Dynamic Payload type for this codec. Valid range is 96–127.	107
Codec Profile X – Loopback Primer (VoiceService.1.VoiceProfile.1.Line.n.Codec.X_LoopbackPrimer.) n = 1 or 2 corresponding to X = A or B, respectively		
Codec	Codec name. The device uses this codec when it acts as a media loopback mirror and before receiving any packets from the loopback source during a media loopback call.	loopbkprimer
PayloadType	Dynamic Payload type for this codec. Valid range is 96–127.	108
Codec Profile X – Codec Settings (VoiceService.1.VoiceProfile.1.Line.n.Codec.X_Settings.) n = 1 or 2 corresponding to X = A or B, respectively		
G726BitPacking	Two values to choose from: - big-endian - little-endian	big-endian

Tone Settings

The Tone Profile parameters web pages include one page for each tone profile.

Tone Profile X Web Page (X = A, B)

The following configuration parameters are available on each page.

Tone Profile A & B Parameter Guide

Parameter	Description	Default Setting
Tone Profile X – Dial Tone (VoiceService.1.VoiceProfile.n.Tone.Description.1.) for n = 1 or 2 corresponding to X = A or B, respectively		
ToneName	Dial Tone.	Dial Tone
TonePattern	Poly Tone Pattern Script.	350-18,440-18;20
Tone Profile X – Ringback Tone (VoiceService.1.VoiceProfile.n.Tone.Description.2.) for n = 1 or 2 corresponding to X = A or B, respectively		

Tone Profile A & B Parameter Guide

Parameter	Description	Default Setting
ToneName	Ringback Tone.	Ringback Tone
TonePattern	Poly Tone Pattern Script.	440-18,480-18;-1;(2+4)

Tone Profile X – Busy Tone (VoiceService.1.VoiceProfile.n.Tone.Description.3.) for n = 1 or 2 corresponding to X = A or B, respectively

ToneName	Busy Tone.	Busy Tone
TonePattern	Poly Tone Pattern Script.	480-18,620-18;10;(.5+.5)

Tone Profile X – Reorder Tone (VoiceService.1.VoiceProfile.n.Tone.Description.4.) for n = 1 or 2 corresponding to X = A or B, respectively

ToneName	Reorder tone or Fastbusy Tone.	Reorder or Fastbusy Tone
TonePattern	Poly Tone Pattern Script.	480-18,620-18;10;(.25+.25)

Tone Profile X – Confirmation Tone (VoiceService.1.VoiceProfile.n.Tone.Description.5.) for n = 1 or 2 corresponding to X = A or B, respectively

ToneName	Confirmation Tone.	Confirmation Tone
TonePattern	Obihai Tone Pattern Script.	600-18;1;(.2+.2)

Tone Profile X – Holding Tone (VoiceService.1.VoiceProfile.n.Tone.Description.6.) for n = 1 or 2 corresponding to X = A or B, respectively

ToneName	Holding Tone played when peer holding the call.	Holding Tone
TonePattern	Poly Tone Pattern Script.	800-18;30;(.1+10)

Tone Profile X – Second Dial Tone (VoiceService.1.VoiceProfile.n.Tone.Description.7.) for n = 1 or 2 corresponding to X = A or B, respectively

ToneName	Second Dial Tone played when dialing second call in a 3-way call.	Second Dial Tone
TonePattern	Poly Tone Pattern Script.	385-18,484-18;20 (n = 1) 400-18,425-18;20 (n = 2)

Tone Profile X – Stutter Dial Tone (VoiceService.1.VoiceProfile.n.Tone.Description.8.) for n = 1 or 2 corresponding to X = A or B, respectively

ToneName	Stutter Dial Tone.	Stutter Dial Tone
TonePattern	Poly Tone Pattern Script.	350-18,440-18;20;2(.1+.1);() (n = 1) 400-18,425-18,450-18;20;2(.1+.04);() (n = 2)

Tone Profile X – Howling Tone (VoiceService.1.VoiceProfile.n.Tone.Description.9.) for n = 1 or 2 corresponding to X = A or B, respectively

ToneName	Howling Tone for off-hook warning.	Howling Tone
-----------------	------------------------------------	--------------

Tone Profile A & B Parameter Guide

Parameter	Description	Default Setting
TonePattern	Poly Tone Pattern Script.	480+3,620+3;10; (.125+.125)
<i>Tone Profile X – Prompt Tone (VoiceService.1.VoiceProfile.n.Tone.Description.10.) for n = 1 or 2 corresponding to X = A or B, respectively</i>		
ToneName	Prompt Tone to prompt user to enter a number for configuration, such as speed dial.	Prompt Tone
TonePattern	Poly Tone Pattern Script.	480-16;20
<i>Tone Profile X – Call Forwarded Dial Tone (VoiceService.1.VoiceProfile.n.Tone.Description.11.) for n = 1 or 2 corresponding to X = A or B, respectively</i>		
ToneName	Call Forwarded Dial Tone: A special dial tone to indicate call-forward-all is active.	Call Forwarded Dial Tone
TonePattern	Poly Tone Pattern Script.	350-18,440-18;20; (.2+.2)
<i>Tone Profile X – Conference Tone (VoiceService.1.VoiceProfile.n.Tone.Description.12.) for n = 1 or 2 corresponding to X = A or B, respectively</i>		
ToneName	Conference Tone (indicates a 3-way conference call has started).	Conference Tone
TonePattern	Poly Tone Pattern Script.	350-16;10; (.1+.1, .1+.9.7) (n = 1) 425-16;10; (1+15, .36+15) (n = 2)
<i>Tone Profile X – SIT Tone 1 (VoiceService.1.VoiceProfile.n.Tone.Description.13.) for n = 1 or 2 corresponding to X = A or B, respectively</i>		
ToneName	Special Information Tone 1.	SIT Tone 1
TonePattern	Poly Tone Pattern Script.	985-16,1428-16,1777-16;20; (1/.380+0, 2/.380+0, 4/.380+0, 0/0+4) (n = 1) 425-16;20; (2.5+.5) (n = 2)
<i>Tone Profile X – SIT Tone 2 (VoiceService.1.VoiceProfile.n.Tone.Description.14.) for n = 1 or 2 corresponding to X = A or B, respectively</i>		
ToneName	Special Information Tone 2.	SIT Tone 2
TonePattern	Poly Tone Pattern Script.	914-16,1371-16,1777-16;20; (1/.274+0, 2/.274+0, 4/.380+0, 0/0+4)
<i>Tone Profile X – SIT Tone 3 (VoiceService.1.VoiceProfile.n.Tone.Description.15.) for n = 1 or 2 corresponding to X = A or B, respectively</i>		
ToneName	Special Information Tone 3.	SIT Tone 3
TonePattern	Poly Tone Pattern Script.	914-16,1371-16,1777-16;20; (1/.380+0, 2/.380+0, 4/.380+0, 0/0+4)

Tone Profile X – SIT Tone 4 (VoiceService.1.VoiceProfile.n.Tone.Description.16.) for n = 1 or 2 corresponding to X = A or B, respectively

Tone Profile A & B Parameter Guide

Parameter	Description	Default Setting
ToneName	Special Information Tone 4.	SIT Tone 4
TonePattern	Poly Tone Pattern Script.	985-16,1371-16,1777-16;20;(1/.380+0,2/.380+0,4/.380+0,0/0+4)

Tone Profile X – Outside Dial Tone (VoiceService.1.VoiceProfile.n.Tone.Description.17.) for n = 1 or 2 corresponding to X = A or B, respectively

ToneName	Outside Dial Tone.	Outside Dial Tone
TonePattern	Obihai Tone Pattern Script.	385-16;10

Tone Profile X – R-Command Tone (VoiceService.1.VoiceProfile.n.Tone.Description.18.) for n = 1 or 2 corresponding to X = A or B, respectively

ToneName	R-Command Tone.	R-Command Tone
TonePattern	Obihai Tone Pattern Script.	400-16;5

Tone Profile X – Paging Tone (VoiceService.1.VoiceProfile.n.Tone.Description.19.) for n = 1 or 2 corresponding to X = A or B, respectively

ToneName	Paging Tone.	Paging Tone
TonePattern	Obihai Tone Pattern Script.	480-16;1;(.2+.2)

Ring Settings

The Ring Profile parameters web pages include one page for each ring profile.

Ring Profile X Web Page (X = A, B)

The following configuration parameters are available on each page.

Ring Profile A & B Parameter Guide

Parameter	Description	Default Setting
<i>Ring Profile X – Call Waiting Tone 1 (VoiceService.1.VoiceProfile.1.Tone.Description.21.)</i>		
ToneName	Distinctive Call Waiting Tone 1.	Bellcore-dr1
TonePattern	Obihai Tone Pattern Script.	440-18;30;(.25+10) (n = 1) 425-18;30;(.2+.2,.2+4.4) (n = 2)
<i>Ring Profile X – Call Waiting Tone 2 (VoiceService.1.VoiceProfile.1.Tone.Description.22.)</i>		
ToneName	Distinctive Call Waiting Tone 2.	Bellcore-dr2
TonePattern	Obihai Tone Pattern Script.	440-18;30;(.1+.1,.3+.1,.1+10)

Ring Profile A & B Parameter Guide

Parameter	Description	Default Setting
<i>Ring Profile X – Call Waiting Tone 3 (VoiceService.1.VoiceProfile.1.Tone.Description.23.)</i>		
ToneName	Distinctive Call Waiting Tone 3.	Bellcore-dr3
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.1+.1, .1+10)
<i>Ring Profile X – Call Waiting Tone 4 (VoiceService.1.VoiceProfile.1.Tone.Description.24.)</i>		
ToneName	Distinctive Call Waiting Tone 4.	Bellcore-dr4
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.1+.1, .1+.1, .1+10)
<i>Ring Profile X – Call Waiting Tone 5 (VoiceService.1.VoiceProfile.1.Tone.Description.25.)</i>		
ToneName	Distinctive Call Waiting Tone 5.	Bellcore-dr5
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.3+.1, .1+.1, .3+10)
<i>Ring Profile X – Call Waiting Tone 6 (VoiceService.1.VoiceProfile.1.Tone.Description.26.)</i>		
ToneName	Distinctive Call Waiting Tone 6.	User-dr1
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.1+.1, .3+.2, .3+10)
<i>Ring Profile X – Call Waiting Tone 7 (VoiceService.1.VoiceProfile.1.Tone.Description.27.)</i>		
ToneName	Distinctive Call Waiting Tone 7.	User-dr2
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.3+.1, .3+.1, .1+10)
<i>Ring Profile X – Call Waiting Tone 8 (VoiceService.1.VoiceProfile.1.Tone.Description.28.)</i>		
ToneName	Distinctive Call Waiting Tone 8.	User-dr3
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.3+2)
<i>Ring Profile X – Call Waiting Tone 9 (VoiceService.1.VoiceProfile.1.Tone.Description.29.)</i>		
ToneName	Distinctive Call Waiting Tone 9.	User-dr4
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.3+2)
<i>Ring Profile X – Call Waiting Tone 10 (VoiceService.1.VoiceProfile.1.Tone.Description.30.)</i>		
ToneName	Distinctive Call Waiting Tone 10.	User-dr5
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.3+2)

Star Codes

The Star Codes parameters web pages include one page for each star code profile.

Star Code Profile X Web Page (X = A, B)

The following configuration parameters are available on each page.

Star Code Profile Parameter Guide

Parameter	Description	Default Setting
Star Code Profile X – Star Codes (VoiceService.1.X_StarCode.n.) for n = 1, 2,..8 corresponding to X = A, B, ..., H		
Code1	Default = Redial Star Code	*07, Redial, call(\$Ldn)
Code2	Default = Call Return Star Code	*69, Call Return, call(\$Lcn)
Code3	Default = Block Caller ID (Persistent) Star Code	*81, Block Caller ID, set(\$Bci,1)
Code4	Default = Unblock Caller ID (Persistent) Star Code	*82, Unblock Caller ID, set(\$Bci,0)
Code5	Default = Block Caller ID Once Star Code	*67, Block Caller ID Once, set(\$Bci1,1)
Code6	Default = Unblock Caller ID Once Star Code	*68, Unblock Caller ID Once, set(\$Ubcil,1)
Code7	Default = Call Forward Unconditional Star Code	*72, Cfwd All, coll(\$Cfan), set(\$Cfa,1)
Code8	Default = Disable Call Forward Unconditional Star Code	*73, Disable Cfwd All, set(\$Cfa, 0)
Code9	Default = Call Forward on Busy Star Code	*60, Cfwd Busy, coll(\$Cfbn), set(\$Cfb,1)
Code10	Default = Disable Call Forward on Busy Star Code	*61, Disable Cfwd Busy, set(\$Cfb, 0)
Code11	Default = Call Forward on No Answer Star Code	*62, Cfwd No Ans, coll(\$Cfnn), set(\$Cfn,1)
Code12	Default = Disable Call Forward on No Answer Star Code	*63, Disable Cfwd No Ans, set(\$Cfn,0)
Code13	Default = Block Anonymous Calls Star Code	*77, Block Anonymous Call, set(\$Bac,1)
Code14	Default = Unblock Anonymous Calls Star Code	*87, Unblock Anonymous Call, set(\$Bac,0)
Code15	Default = Enable Call Waiting Star Code	*56, Enable Call Waiting, set(\$Cwa,1)
Code16	Default = Disable Call Waiting Star Code	*57, Disable Call Waiting, set(\$Cwa,0)
Code17	Default = Do Not Disturb Star Code	*78, Do Not Disturb, set(\$Dnd,1)
Code18	Default = Disable Do Not Disturb Star Code	*79, Disable DND, set(\$Dnd,0)

Star Code Profile Parameter Guide

Parameter	Description	Default Setting
Code19	Default = Repeat Dial Star Code	*66, Repeat Dial, rpdi(\$Ldn)
Code20	Default = Disable Repeat Dial Star Code	*86, Cancel Repeat Dial, rpdi()
Code21	Default = Set Speed Dial Star Code	*74([1-9] [1-9]x), Set Speed Dial, coll(\$Spd[\$Code])
Code22	Default = Check Speed Dial Star Code	*75([1-9] [1-9]x), Check Speed Dial, say(\$Spd[\$Code])
Code23	Default = Loopback Media Star Code	*03, Loopback Media, set(\$Lbm1,1)
Code24	Default = Loopback RTP Star Code	*04, Loopback RTP Packet, set(\$Lbp1,1)
Code25	Default = Force G711u Codec Star Code	*4711, Use G711 Only, set(\$Cdm1,3)
Code26	Default = Force G729 Codec Star Code	*4729, Use G729 Only, set(\$Cdm1,4)
Code27	Default = Clear Speed Dial Star Code	*76([1-9] [1-9]x), Clear Speed Dial, set(\$Spd[\$Code],)
Code28	Default = Blind Transfer Star Code	*98, Blind Transfer, coll(\$Bxrn)
Code29	Default = Barge In Star Code	*96, Barge In, set(\$Bar1,1)
Code30		
Code31		
Code32		
Code33	Default = Force OPUS Codec Star Code	*4678, Use OPUS Only, set(\$Cdm1,1024)

User Settings

The User Settings parameters web pages include the following pages:

- [Speed Dial Settings](#)
- [User-Defined Digit Maps Settings](#)

Speed Dial Settings

The following configuration parameters are available on this page.

Speed Dial Parameter Guide

Parameter	Description	Default Setting
<i>Speed Dials (SpeedDial.)</i>		
1	Enter the number or the URL for Speed Dial 1 with an associated name.	
2	Enter the number or the URL for Speed Dial 2 with an associated name.	
3	Enter the number or the URL for Speed Dial 3 with an associated name.	
...	...	
99	Enter the number or the URL for Speed Dial 99 with an associated name.	

User-Defined Digit Maps Settings

For an explanation of this feature, refer to the [User-Defined Digit Maps](#) section in the [Device Settings](#) chapter.

User-Defined Digit Maps Parameter Guide

Parameter	Description	Default Setting
User-Defined Digit Map 1 (<i>UserDigitMap.1.</i>)		
Label	A 2- to 16-character long label to reference this digit map in other digit maps and call routing rules. It must be alphanumeric, not contain any spaces, and be different from other user-defined or built-in digit map labels.	ipd
DigitMap	A valid digit map.	(xx.<*: @>xx?x?<*: .> xx?x?<*: .>xx?x?<*: . >xx?x? xx.<*: @>xx?x ?<*: .>xx?x?<*: .>xx? x?<*: .>xx?x?<*: :>xx ?x?x?x?) Note: This default value supports IPv4 dialing.
User-Defined Digit Map N (<i>N = 2 to 10</i>)		
Label	A 2- to 16-character long label to reference this digit map in other digit maps and call routing rules. It must be alphanumeric, not contain any spaces, and be different from other user-defined or built-in digit map labels.	
DigitMap	A valid digit map.	