MOTOTRBO™ Development Specification -

Network Application Interface Voice Dispatch And Control Signaling Services

Version 01.02 5/15/2014

COPYRIGHTS

REVISION HISTORY

Date	Version	Feature	Section	Page	Lines	Description
June 18, 2013	0100	All	All	All	All	Initial Draft
Dec 10, 2013	0101	Link Establishment	3.1	41	687-689	Specify the application setting for “No LE Connection to Data Revert Repeater”
		Wireline Registration	3.2	53-54 54-55 55	856-878 887-895 903-920	Add example wireline registration setting for different applications
		Voice Call Setup	3.5.9.5	112-113	1636-1639	Add message sequence to handle Lost Response
			3.5.9.6	114	1656-1661	Clarify how to handle lost WL_VC_CALL_SESSION_STATUS message
			3.5.9.7	114-116	1662-1685	Clarify the handling of Call Session End message during call setup
			3.5.9.10	117	1707-1709	Clarify the limitation on supporting console audio loop back in R2.3
			5.2	161	2469-2470	Clarify how soon the application shall send the CSBK ACK
			6.8.27	197-201	2914	Based on the declined reason, specify how long the application shall wait before retrying.
		Wireline Protocol Version	6.7.11.3 6.7.13.2 6.7.14.2	181 185 186	2725-2728 2751-2754 2763-2766	Clarify the application shall not check the protocol version at the WL_VC_VOICE_END_BURST, WL_VC_VOICE_BURST, WL_VC_VOICE_PRIVACY_BURST

		Information Field Details	6.8.7	190	2811-2813	Clarify the relationship between the Address Type and the Start/End Address Range
			6.8.8	190	2818-2820	Clarify the bit settings in Voice Attribute field
			6.8.9	190	2825-2827	Clarify the bit settings in CSBK Attribute field
Apr 24, 2014	0101	Error Fix	3.6.5	134	1922, 1925	Remove the extra 2-byte in the CSBK message examples (CCMP01884889)

TABLE OF CONTENTS

1.0 Overview 9
1.1 Purpose 9
1.2 Scope of the Document 9
1.3 Assumption 10
1.4 Terminology 10
1.5 Reference 12
2.0 System Overview 14
2.1 Single Site Conventional System Overview 14
2.2 IP Site Connect System Overview 15
2.3 Capacity Plus System Overview 17
2.4 Linked Capacity Plus System Overview 20
2.5 Call Control Interface Overview 25
2.5.1 Call Control Interface Capability 28
2.5.2 Interface Comparison 30
2.6 Interface to Device Detection and Mobility Management (DDMS) 33
2.6.1 Route Information 33
2.6.2 Interface to DDMS 34
2.7 Example System Topologies 36
2.7.1 Voice and Data Application Server 36
2.7.2 Multiple IP Site Connect Systems with Centralized Application Server 36
2.7.3 Wide and Local Area IP Site Connect Systems with Centralized Application Server 38
2.7.4 Centralized Application Server for Multiple Systems 39
3.0 Call Control Interface Protocol 41
3.1 Peer Discovery through Link Establishment 41
3.1.1 Setting in the LE messages 44
3.1.2 Staggering LE Keep Alive Message in LCP System 50
3.2 Wireline Registration 51
3.2.1 Call Routing based on Wireline Registration. 56
3.2.2 Initial Wireline Registration 58
3.2.3 Wireline Registration Change 64
3.2.4 Wireline Deregistration 65
3.2.5 Wireline Registration in Single Site / IPSC System 66
3.2.6 Wireline Registration in Capacity Plus System 67
3.2.7 Wireline Registration in Linked Capacity Plus System 68
3.3 Wireline Protocol Version 69
3.3.1 Wireline Protocol Version Definition 70
3.3.2 Wireline Protocol Version Negotiation 70
3.4 Wireline Authentication 71
3.4.1 Wireline Authentication Footer 71
3.4.2 Wireline Authentication Key and Signature 72
3.5 Wireline Voice Call 75
3.5.1 Call Stream and Call Session 75
3.5.2 Roles in the Voice Call 76
3.5.3 Call Arbitration 78

3.5.4 Audio Data Format ... 80
3.5.5 RTP Header for the Call Control Voice Packet ... 83
3.5.6 Example Call Control Voice Packets ... 84
3.5.7 Example Voice Call Message Sequences ... 86
3.5.8 Timing at Network Interface ... 106
3.5.9 Exception Handling ... 107
3.5.10 Call Access Criteria ... 114
3.5.11 Impolite Access Over On-Going Call ... 116
3.5.12 Transmit Interrupt ... 118
3.6 Wireline Control Calls (CSBK) ... 127
3.6.1 Call Session ... 127
3.6.2 Call Routing ... 127
3.6.3 Roles in the CSBK Call ... 127
3.6.4 DMR CSBK Bursts ... 128
3.6.5 Wireline CSBK Messages ... 129
3.6.6 Preamble CSBK ... 131
3.6.7 Exception Handling ... 131
3.6.8 CSBK Call Example Message Sequence ... 132
3.6.9 IP Console Inhibit (Application to Radio) ... 138
4 Voice Privacy ... 141
4.1 Basic Privacy ... 141
4.1.1 Privacy Processing ... 141
4.1.2 Voice Packet for Basic Privacy ... 143
4.2 Enhanced Privacy ... 144
4.2.1 Privacy Processing ... 144
4.2.2 Privacy Algorithms ... 147
4.2.3 Message Settings for Enhanced Privacy ... 148
5 System Configuration ... 155
5.1 Linked Capacity Plus / Capacity Plus Rest Channel IP Address / UDP Port ... 155
5.2 Call Hang Time Configuration ... 156
5.3 Third Party Application Configuration List ... 158
6 Call Control Protocol Definition ... 159
6.1 Call Control PDU Structure ... 159
6.2 Wireline Authentication Footer ... 159
6.3 Message Direction ... 160
6.4 Byte Order ... 160
6.5 Wireline Opcode ... 160
6.6 Key to message Specification ... 161
6.6.1 Message Dashboard ... 161
6.6.2 Message Field Types ... 161
6.6.3 Reserved Fields ... 162
6.6.4 Packet Format per Call Control Protocol Version ... 162
6.7 Call Control Message Definition ... 162
6.7.1 Basic Message Format ... 162
6.7.2 0x01– WL_REGISTRATION_REQUEST ... 164

6.7.3 0x02– WL_REGISTRATION_ GENERAL_OPS 166
6.7.4 0x03– WL_REGISTRATION_ STATUS 167
6.7.5 WL_CHNL_STATUS 168
6.7.6 0x12– WL_CHNL_STATUS_QUERY 170
6.7.7 0x13– WL_VC_CHNL_CTRL_REQUEST 171
6.7.8 0x16 – WL_VC_CHNL_CTRL_STATUS 172
6.7.9 0x17 – WL_VC_CSBK_CALL 173
6.7.10 0x18 –WL_VC_VOICE_START 174
6.7.11 0x19 – WL_VC_VOICE_END_BURST 175
6.7.12 0x20 – WL_VC_CALL_SESSION_STATUS 177
6.7.13 0x21– WL_VC_VOICE_BURST 178
6.7.14 0x22– WL_VC_PRIVACY_BURST 180
6.8 Information Field Details 182
6.8.1 Peer ID 182
6.8.2 Registration Slot Number 182
6.8.3 Registration PDU ID 182
6.8.4 Registration ID 183
6.8.5 Wireline Channel Status Flag 183
6.8.6 Address Type 183
6.8.7 Start/End of Register Addresses 184
6.8.8 Voice Attributes 184
6.8.9 CSBK Attributes 184
6.8.10 Wireline Protocol Version 184
6.8.11 Registration Operation Code 185
6.8.12 Registration Status 185
6.8.13 Registration Status Code 185
6.8.14 Status PDU ID 185
6.8.15 Slot number 185
6.8.16 Channel Status 186
6.8.17 Call ID 187
6.8.18 Source ID 187
6.8.19 Target ID 187
6.8.20 Call Type 187
6.8.21 Call Session Status 189
6.8.22 Access Criteria 189
6.8.23 Call Attributes 190
6.8.24 Preamble Duration 190
6.8.25 CSBK Arguments 190
6.8.26 Channel Control Request Status 190
6.8.27 Decline Reason Code 191
6.8.28 Manufacturer's ID (MFID) 196
6.8.29 RTP Header Information Field 196
6.8.30 Burst Type 198
6.8.31 Service Option 198
6.8.32 AMBE Encoded Voice Frames 198

6.8.33 Raw RSSI Information 199
6.8.34 Algorithm ID, Key ID and IV 200
7 Motorola Solutions Proprietary Message Definitions 202
7.1 DMR CSBK Messages 202
7.1.1 Single Block Packet Description 202
7.1.2 Acknowledge Response - Unit (ACK_RSP_U) 204
7.1.3 Call Alert Request (CALL_ALERT_REQ) 204
7.1.4 Emergency Alarm Request (EMRG_ALRM_REQ) 205
7.1.5 Radio Remote Monitor command (RAD_MON_CMD): 205
7.1.6 Extended Function Command (EXT_FNCT_CMD) 206
7.1.7 Extended Function Response (EXT_FNCT_RSP) 207
7.1.8 Field Definitions 208
Appendix A: LFSR Sample Implementation 210
Appendix B: Golay(24, 12, 8) Matrix 211
Appendix C: 212

1.0 Overview

1.1 Purpose

Due to a lot of limitations on the existing IP Site Connect Protocol, starting from R2.2A the MOTOTRBO repeater supports a new IP based Network Application Interface to enable the bi-direction voice/data/CSBK call between third party applications and MOTOTRBO radios. Through this interface a third party application can connect to the MOTOTRBO digital repeater system through an IP connection instead of a control station.

The MOTOTRBO repeater Network Application Interface has the following components:

- Link Establishment Interface: Support MOTOTRBO repeaters or third party applications to join in the MOTOTRBO repeater system.
- Data Call Interface: Motorola Solutions provides a Windows PC based service application called MOTOTRBO Network Interface Service (MNIS) to support the data calls between the repeater network based application and MOTOTRBO radios. Please refer to MOTOTRBO Network Interface Service ADK for further information.
- Call Control Interface: Support voice / CSBK calls between the repeater network based application and MOTOTRBO radios.
- Repeater Call Monitoring Interface: Support call logging messages to the repeater network based application.
- XCMP/XNL Interface: Support XCMP/XNL messages to provide Repeater Diagnostic, Alarm and Control (RDAC) function for the repeater network based application.

There are separate ADK documents to describe each of the interfaces. This document only describes the Call Control interface, including the specification of the protocol and the functionality of the protocol commands and responses.

1.2 Scope of the Document

The Network Application Interface based applications still need to follow the existing Link Establishment (LE) protocol to join into the repeater system. This document only covers the Call Control messaging for voice / CSBK calls after the LE connection. Refer to the MOTOTRBO Link Establishment Protocol Specification for the LE connection.

For call monitoring service, refer to MOTOTRBO Repeater Call Monitoring Protocol Specification.

For Repeater Diagnostics, Alarm and Control (RDAC) function, refer to MOTOTRBO Repeater XCMP Development Guide and Specification.

1.3 *Assumption*

The reader of the document is assumed have the following domain knowledge:

- Principle of two-way radio communications
- ETSI Digital Mobile Radio (DMR) Air Interface Protocol
- Open Systems Interconnection (OSI) Model
- Real-time Transport Protocol (RTP)
- UDP/IP Protocol
- MOTOTRBO Linked Establishment (LE) Protocol

The following domain knowledge is considered beneficial, but is not required:

- Digital two-way radio communications
- Time division multiplexing (TDM)
- Voice Encoding/Decoding

1.4 *Terminology*

Ack - Acknowledgement

AMBE – Advanced Multi-Band Excitation

ARP – Address Resolution Protocol

CAI – Common Air Interface

CPC – Capacity Plus

CPS - Customer Programming Software

CRC – Cyclic Redundancy Checksum for data error correction

CSBK – Control Signaling BlocK

CWID - Continuous Waveform Identifier

DDMS – Device Discovery and Mobility Service

DMR - Digital Mobile Radio

DVSI – Digital Voice Systems, Inc

ETSI - European Telecommunications and Standards Institute

FCC – U.S. Federal Communications Commission

FEC – Forward Error Correction

FID – Feature set ID

ID - IDentifier

IP – Internet Protocol

IPSC – IP Site Connect

IPv4 – IP version 4

ISP - Internet Service Provider

LAC – Local Area Channel

LC – Link Control

LCP – Linked Capacity Plus

LE – Link Establishment

MCDD – Multi-Channel Device Driver

MFID – Manufacturer's FID

MNIS – MOTOTRBO Network Interface Service

Nack – Negative Acknowledgement

OACSU – Over-the-Air Call SetUp

PATCS – Push And Talk Call Setup

PDU – Protocol Data Unit

PTT – Push To Talk

RCM – Repeater Call Monitoring

RDAC - Repeater Diagnostics, Alarm and Control

RF – Radio Frequency

RSSI - Received Signal Strength Indication

RTP – Real-time Transport protocol

SS – Single Site system

SU – Subscriber Unit

TDMA – Time Division Multiple Access

UDP – User Datagram Protocol

USB – Universal Serial Bus

WAC – Wide Area Channel

XCMP – eXtended Control and Management Protocol

XNL – XCMP Network Layer protocol

1.5 *Reference*

[1] MOTOTRBO Link Establishment Protocol Specification

[2] MOTOTRBO Repeater XCMP Protocol Specification

[3] MOTOTRBO Repeater XCMP Development Guide

[4] MOTOTRBO Data Service Overview

[5] MOTOTRBO Repeater Call Monitoring Protocol Specification

[6] MOTOTRBO Device Discovery and Mobility Service-to-Watcher Interface Protocol Specification

[7] MOTOTRBO Network Interface Service ADK Development Guide

[8] Electromagnetic compatibility and Radio spectrum Matters (ERM; Digital Mobile Radio (DMR) Systems; Part 1: DMR Air Interface (AI) protocol, ETSI TS 102 361-1 V1.4.5 (2007 -12)

[9] Electromagnetic compatibility and Radio spectrum Matters (ERM; Digital Mobile Radio (DMR) Systems; Part 2: DMR voice and generic services and facilities, TS 102 361-2 V1.2.6 (2007-12)

[10] Electromagnetic compatibility and Radio spectrum Matters (ERM; Digital Mobile Radio (DMR) Systems; Part 3: Packet data protocol, TS 102 361-3 V1.1.7 (2007-12)

[11] ARC4: http://en.wikipedia.org/wiki/RC4

[12] MOTOTRBO System Planner

[13] MOTOTRBO™ XCMP/XNL Development Guide

[14] MOTOTRBO Customer Programming Software on-line help

[15] RFC 3174 – US Secure Hash Algorithm (SHA-1), September 2001, D. Eastlake, 3rd, http://www.faqs.org/rfcs/rfc3174.html

[16] RFC 3550 RTP: A Transport Protocol for Real Time Application, H. Schulzrinne, July 2003 http://tools.ietf.org/html/rfc3550

2.0 System Overview

Section 2.1 – Section 2.4 provides a system overview on each of the systems supported by the Call Control interface. Besides describing the basic system operation theory, those sections cover the call types, channel allocation and channel types, which are the fundaments for the understanding of the Call Control interface. If the reader is familiar with those topics, he can skip those sections and start with Section 2.5.

2.1 Single Site Conventional System Overview

The Single Site Conventional system is introduced in MOTOTRBO 1.0 System Release. The repeater operating in Single Site Conventional mode receives the signal from the radio in one frequency and repeats the signal in another frequency after error detection and correction. By doing that the repeater increases the system RF coverage. The repeater supports two TDMA slots or channels in each frequency, which doubles the user capacity, comparing with the talk-around mode. Two simultaneous calls can be supported in one system. As shown in Figure 1, radio1 and radio2 uses slot1 for a call; radio3 and radio4 uses slot2 for another call. Third party applications which connects to the repeater can send/receive voice or CSBK call with any of the radio in the system.

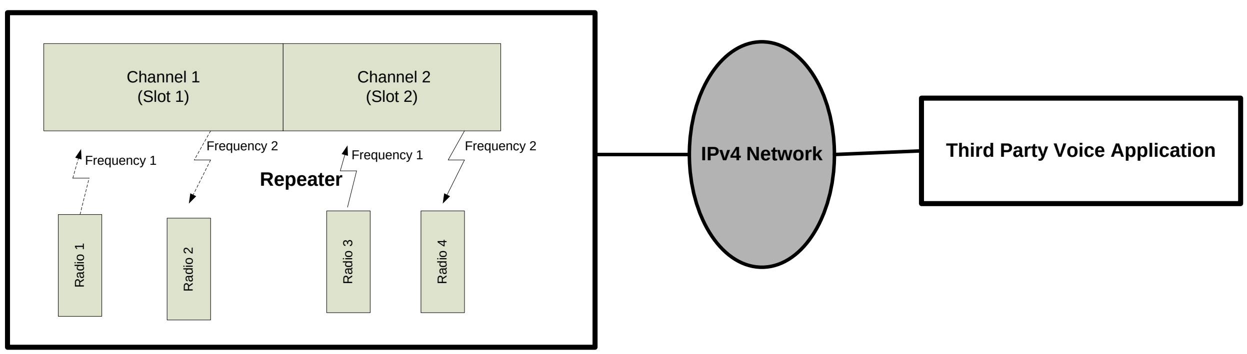


Figure 1: Two Calls in one Single Site Conventional System

To coordinate the channel access from different end users, the repeater announces the channel status. The radio with polite access criteria only starts a call if the channel is idle.

In a voice call, after one user finishes, the other user talks back. The repeater reserves a short duration to allow the other call member to respond by indicating the channel is busy for that call. We call that short duration reserved by the repeater the Call Hang Time. After the Call Hang Time expires, the repeater announces the channel to be idle and starts the Channel Hang Time. All the end users can access the channel while the channel is idle. When the Channel Hang Time expires, if no one is using the channel, the repeater moves to hibernating state. The repeater notifies the third party application on the channel status change. The third party application shall use channel status information to decide when it can initiate a call from the Call Control interface.

In a Single Site system, the radio can configure a dedicated channel to send GPS location data to the location server at each selected channel, called the GPS Revert

Channel. No voice / data / CSBK call can be sent on this channel from the application to the radio or from the radio to the radio.

2.2 *IP Site Connect System Overview*

The IP Site Connect system is introduced in MOTOTRBO 1.4 System Release. In the IP Site Connect mode, the MOTOTRBO digital repeater transmits the received voice / data / CSBK packet over an IP network to each peer in the system. The voice, data and control packets can be exchanged across disperse locations and different radio frequency bands.

For example, the Radio Frequency (RF) coverage is increased when multiple repeaters in the same campus are connected through the IP Site Connect system. Also, voice or data communication across geographically separate locations can be achieved when MOTOTRBO repeaters located in different locations, that could even have different frequencies and/or different color codes, are connected by IP Site Connect.

When the MOTOTRBO Repeater operates in the IP Site Connect mode, it supports two TDMA time slots or logical channels. The Local Area Channel (LAC) is used to repeat the on-site traffic. It has the exact same functions and coverage area as the MOTOTRBO Single Site Conventional Digital Channel. The Wide Area Channel (WAC) is used to repeat the received traffic to the peers which dedicate the same time slot for the same purpose.

WAC is the sum of all wide area time slots that are tied together in the IP Site Connect system. When a call is initiated by a subscriber in the WAC at one site, it is repeated by the repeater at that site across all the wide area channels that are linked together. All the sites transmit at the same time. In one IP Site Connect system, there can be two WACs at most because the time slot number assigned to one WAC must be the same and the repeater can support at most two time slots. The configuration of each channel is independent to each other. When one channel is configured as WAC, the other channel can be either configured as LAC or WAC.

Figure 2 shows an IP Site Connect system with two WACs and one LAC. Radio5 initiates a call in WAC2 at site2. Repeater at site2 sends it to all the connected peers by unicasting the call to each peer. The peers with slot2 configured as WAC repeat the call over the air. Radio3 and radio7 that are in WAC2 receive the call while they are at different sites. When radio1 initiates a call in LAC1 at site1, the repeater at site1 repeats it over the air instead of sending it to all the connected peers. The call is only received by the radios at site1.

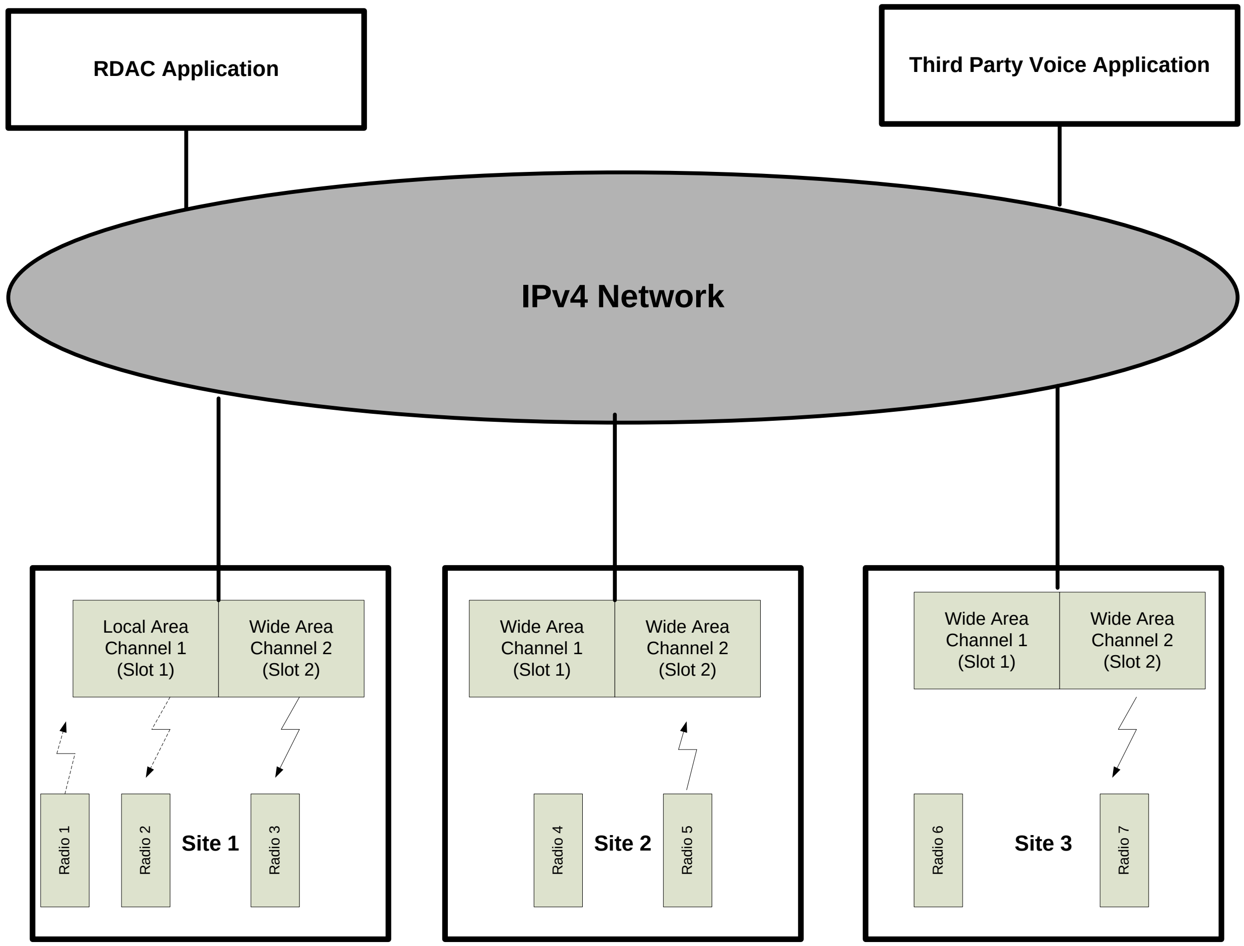


Figure 2: IP Site Connect system with Two Wide Area Channels

The IP Site Connect system increases the RF coverage. The connected WAC in the IP Site Connect system has the same system capacity as the MOTOTRBO single site conventional digital channel.

The repeater sends the over-the-air traffics received at the IP Site Connect wide area slot to all the repeaters and applications in the same IP Site Connect system. The repeater sends the over-the-air traffics received at the IP Site Connect local slot only to the applications in the system. In Figure 2, the same third party voice application can send and receive the voice / CSBK calls from both the wide area slot and the local area slot. The RDAC application can monitor all the repeaters' status in the system.

Similar to the Single Site conventional system, one of the slots in the IP Site Connect repeater can be used for GPS revert channel. The GPS Revert Channel can be configured as a WAC or LAC in IP Site Connect System.

2.3 Capacity Plus System Overview

The Capacity Plus (CPC) system is introduced in MOTOTRBO 1.5 System Release. In the Capacity Plus mode, MOTOTRBO digital repeaters form a single site trunked system, where a pool of channels are shared to support a large group of talkgroups and radio users. The shared channels are called Trunked Channels. While in a conventional radio system, once a channel is selected, there is only one channel to be used for the talkgroups. The system capacity and channel efficiency in the Capacity Plus system is much higher than the conventional systems. The repeaters in Capacity Plus mode join the system using the same Link Establishment Protocol as the repeaters in the IP Site Connect mode. In a trunked radio system, the number of shared channels is twice the number of trunked repeaters.

In a Capacity Plus system, all the “idle” radios which are not receiving or transmitting a call stay on an idle channel, called the Rest Channel. A new call always starts on the Rest Channel. The Rest Channel Repeater, which has the Rest Channel, selects one of the idle channels as the new Rest Channel, and informs all the repeaters and all the “idle” radios about the new Rest Channel. All the radios that are not party to the new call move to the new Rest Channel, and the new call continues at the old Rest Channel, which becomes a traffic channel. At the end of the call (i.e. after the call hang timer times out), the repeater at the current traffic channel sends message over-the-air to inform the current Rest Channel and call status (including Talkgroup Id) of all the channels in the system. The radios can either move to the Rest Channel or to a channel where a group call of interest is in progress.

Figure 3 shows a Capacity Plus system with four Trunked Channels and two Data Revert Channels. Before radio1 initiates a call to radio2, all the “idle” radios (radio1, radio2, radio4 and radio5) are at the Rest Channel of Trunked Channel1. When radio1 initiates the call to radio2, repeater1 selects the Trunked Channel4 as the new Rest Channel. All the radios (radio4 and radio5) that are not part of the call move to the new Rest Channel. When radio1 and radio2 finish the call, they move to the Rest Channel at Trunked Channel4 also.

In Figure 3, the third party voice application can send and receive the voice / CSBK calls from any of the radio in the system. When the third party voice application initiates a call through the Call Control interface, it shall always send the message to the Rest Channel Repeater.

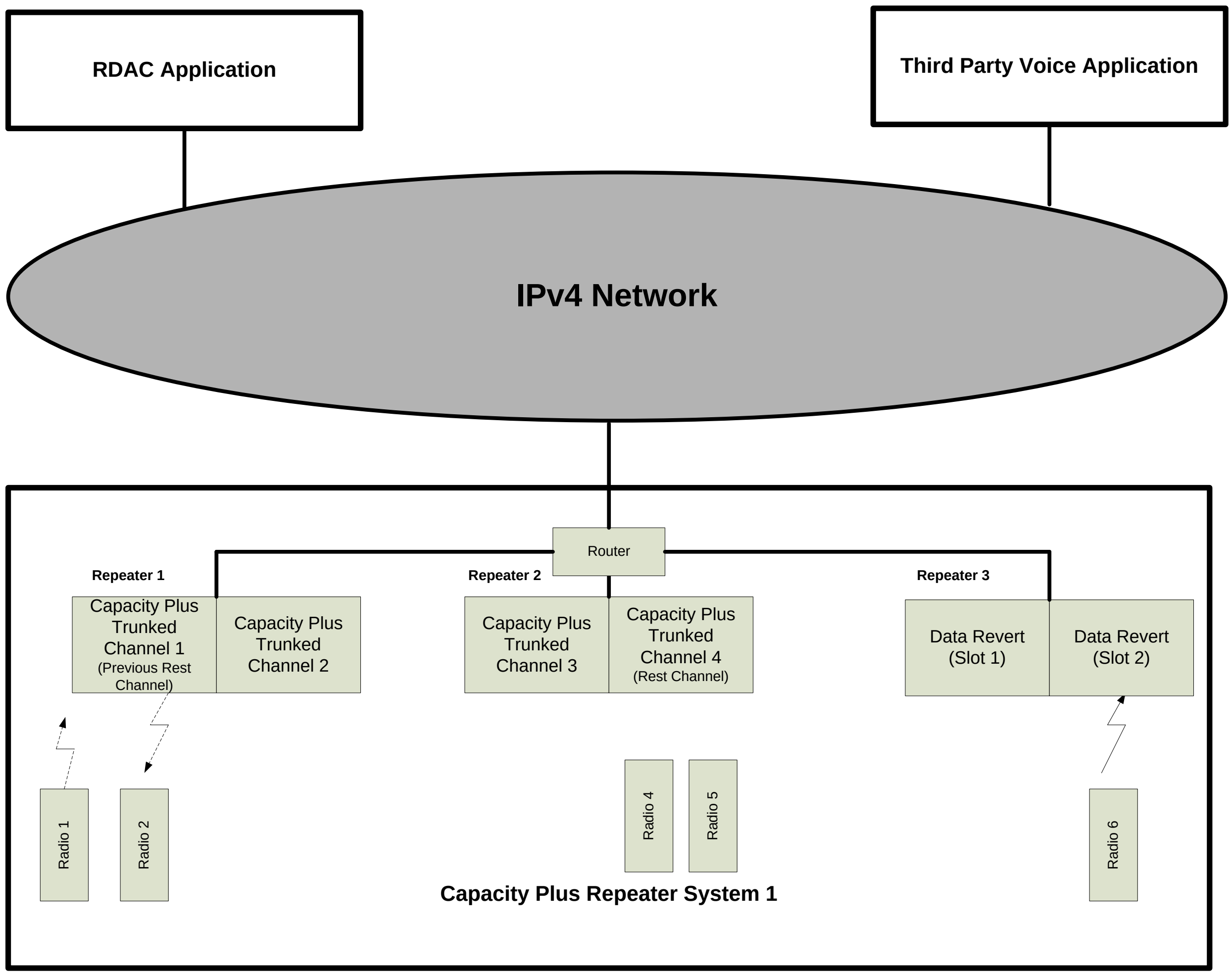


Figure 3: Capacity Plus System with Four Trunked Channels and Two Data Revert Channels

To support more voice communication on the Trunked Channels, the Capacity Plus system supports the non-Trunked Channels called Data Revert Channel to offload most of the data communication from the Trunked Channels. The Data Revert Channels are used by a radio, or third party XCMP device attached to the radio to send data packets (e.g. location responses, Text messages, etc.) to the application data server only. For repeaters in Capacity Plus mode, both channels of the repeater are used for the same purpose. Both channels are either for the Trunked Channels or for the Data Revert Channels.

In Capacity Plus system, all the repeater peers are required in the same Local Area Network (LAN) due to the fast movement of the Rest Channel. It is very difficult for the third party application to follow the Rest Channel movement if it is not in the same LAN as the repeater peers. Even when they are in the same LAN, the tracking of the Rest Channel is complex. To make the third party application interface in Capacity Plus system available in both the repeater's LAN and the Wide Area Network (WAN) topologies, and to make the application transparent to the Rest Channel movement, the repeater peer in the Capacity Plus system supports IP aliasing by associating a static IP address and UDP port with the Rest Channel, shown as IP address of 10.1.2.1 in blue font in Figure 4. This static IP address and UDP port seems to be another peer in the system. We call it the Virtual Peer. Each repeater peer in the Capacity Plus system has its unique IP address and UDP port to join the system, and also supports the IP address of the Virtual Peer. The Capacity Plus system is a single site trunking system. The Virtual Peer is the system access point for application. Therefore, we also call the Virtual Peer as the Site Peer. When the repeater becomes the Rest Channel repeater, it associates its MAC address with the Site Peer's IP address by sending a Gratuitous ARP (Address Resolution Protocol) message to the router and all other repeaters. The router updates the MAC address of the Rest Channel repeater vs. the Site Peer's IP address in its ARP cache. The router uses this map to send the IP message with destination of the Site Peer's IP address to the Rest Channel repeater. When the third party application initiates a call on the Rest Channel, it sends message to the Site Peer's static IP address without actually following the Rest Channel movement as the repeater peers.

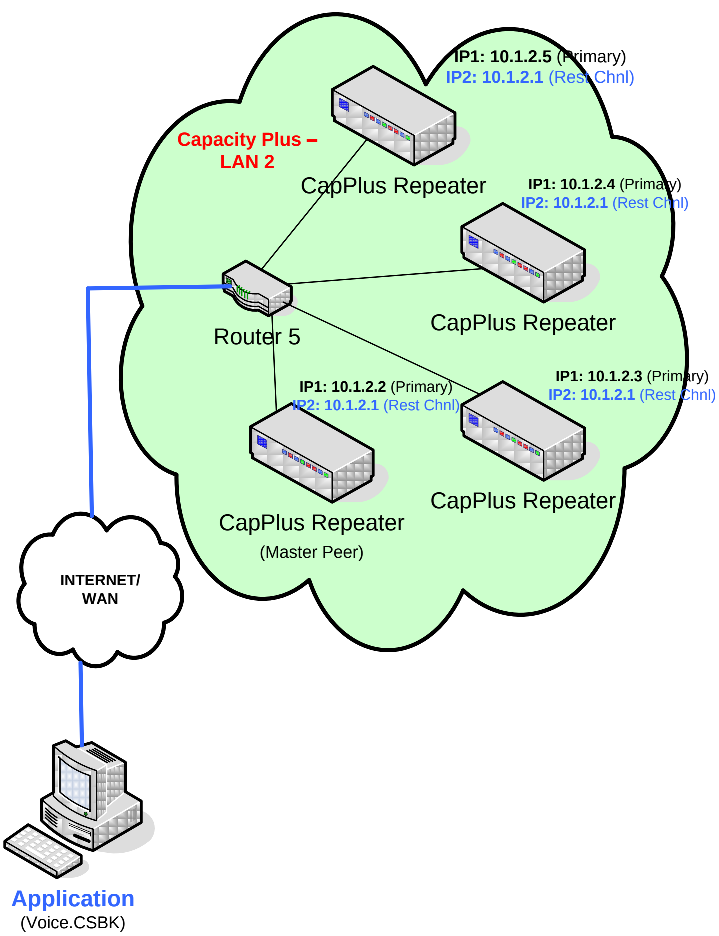


Figure 4: Capacity Plus System with Site Peer to Support Third Party Application

2.4 *Linked Capacity Plus System Overview*

The Linked Capacity Plus (LCP) system is introduced in MOTOTRBO 1.9 System Release. In the Linked Capacity Plus mode, the MOTOTRBO digital repeaters form a multi-channel trunked system. The repeaters determine the Rest Channel in each site as the Capacity Plus system and form a multisite system to distribute the voice and data payload as the IP Site Connect system. Therefore, Linked Capacity Plus system uses both the Capacity Plus system and IP Site Connect system's concepts albeit with modifications and adaptations as below:

- No controller in each site. LCP repeaters at a site are connected over a LAN and the Rest Channel is selected by the current traffic channel repeater as a Capacity Plus System. Similar to Capacity Plus system, LCP repeaters at a site are connected over a LAN. A Capacity Plus repeater uses individual messages to inform all the repeaters at its site, but an LCP repeater uses IP Limited Broadcast Address to distribute a message (e.g. Keep Alive) that need to go to all the repeaters at a site. The trunked repeaters must be plugged into a switch.
- Dynamically forms an IP Site Connect System between the current Rest Channels among the sites to handle the new initiated wide area call.
- A Revert Channel, including Enhanced GPS revert channel, can be either a wide-area channel or a local channel. The wide-area revert channels of different sites form an IP Site Connect system. The local-area revert channel is intended for local communication. For details, refer to Reference [9].
- Same as in the IPSC and Capacity Plus system, there is one repeater with a static IP address acting as the Master peer. The Master Peer maintains the system map. All the repeaters in the LCP system first register with the Master Peer to get the system map and exchange Keep Alive messages to keep firewall open.
- By not dividing the channels into two groups of Local Area Channel (LAC) and Wide Area Channels (WAC), the number of channels available for trunking increases, which improve the trunking efficiency. LCP allows a customer to reserve a number of logical channels (that is, the TDMA slots) for wide area calls only in each site. A wide area call starts only if all the associated sites have idle channels. Thus the reserved channels improve the successful call setting up rate for wide area calls. For more details, refer to Reference [9] section 2.2.1.6.

There are 3 call types in the Linked Capacity Plus System:

- Local Call. Local calls are received by radios only at the site where the call was initiated. All the radios whose software is below the MOTOTRBO 1.9 System Release can only initiate local individual calls or local group calls. For the upgrade and migration detail, refer to Reference [9] Section 4.6.6. Radios with software after the MOTOTRBO 1.9 System Release can use the Local Talkgroup ID to initiate local calls. With LCP, channels are no longer statically defined to be a local channel.

- Wide-area Group Call. Wide-area Talkgroup IDs are defined in MOTOTRBO CPS. A wide-area talkgroup call lights up only the statically associated sites (see Figure 5 for an example). To improve the trunking efficiency, all the trunking channels can be allocated for either local calls or wide-area calls.

- LCP Individual Call. No matter the target radio is in the same site or different site from the source radio, all the sites "lights-up" to search for the target radio and request a special Acknowledge over the air from the target radio. After approx. 420 ms, the call continues only between the two participated sites where the source radio or destination radio are present. No ARS message is needed in this procedure to allocate the target radio.

In a Linked Capacity Plus system, each site may consist of one or more Trunked repeaters and optionally one or more Data Revert repeaters. Detailed descriptions about LCP topologies can be found in Reference [9]. Figure 5 shows an example LCP system. Repeater2 in site1 plays the Master Peer role. Repeater3 in site1, repeater2 in site2, and repeater3 in site3 are all data revert repeaters, which can be either wide-area revert repeaters or local area revert repeaters. All the others are trunked channel repeaters, which means site1 and site3 each has 4 trunked channels and site2 has 2 trunked channels.

In site1, an individual call between radio1 and radio2 is ongoing using repeater1 slot1. When radio3 initiates a wide-area group call, the pre-defined associated site2 and site3, as configured in Master Peer's CPS file, is lighted up and grant their current rest channel (repeater1 slot1 in each site) to repeat the call over the air to the interested radio5 and radio8. In site2, radio6 wants to make a LCP individual call to radio9. Current rest channel in site2 is Repeater1 Slot2, which accepts it and sends "call request" invitation messages to all the sites in the whole system. After arbitration and searching, site3 inform site2 that target radio9 is available in its scope, then the call between radio6 and radio9 is set up and site3 officially grants the repeater2 slot2 to transmit the subsequent voice bursts from site2. In site3, a local group call among radio is ongoing using repeater2 slot1.

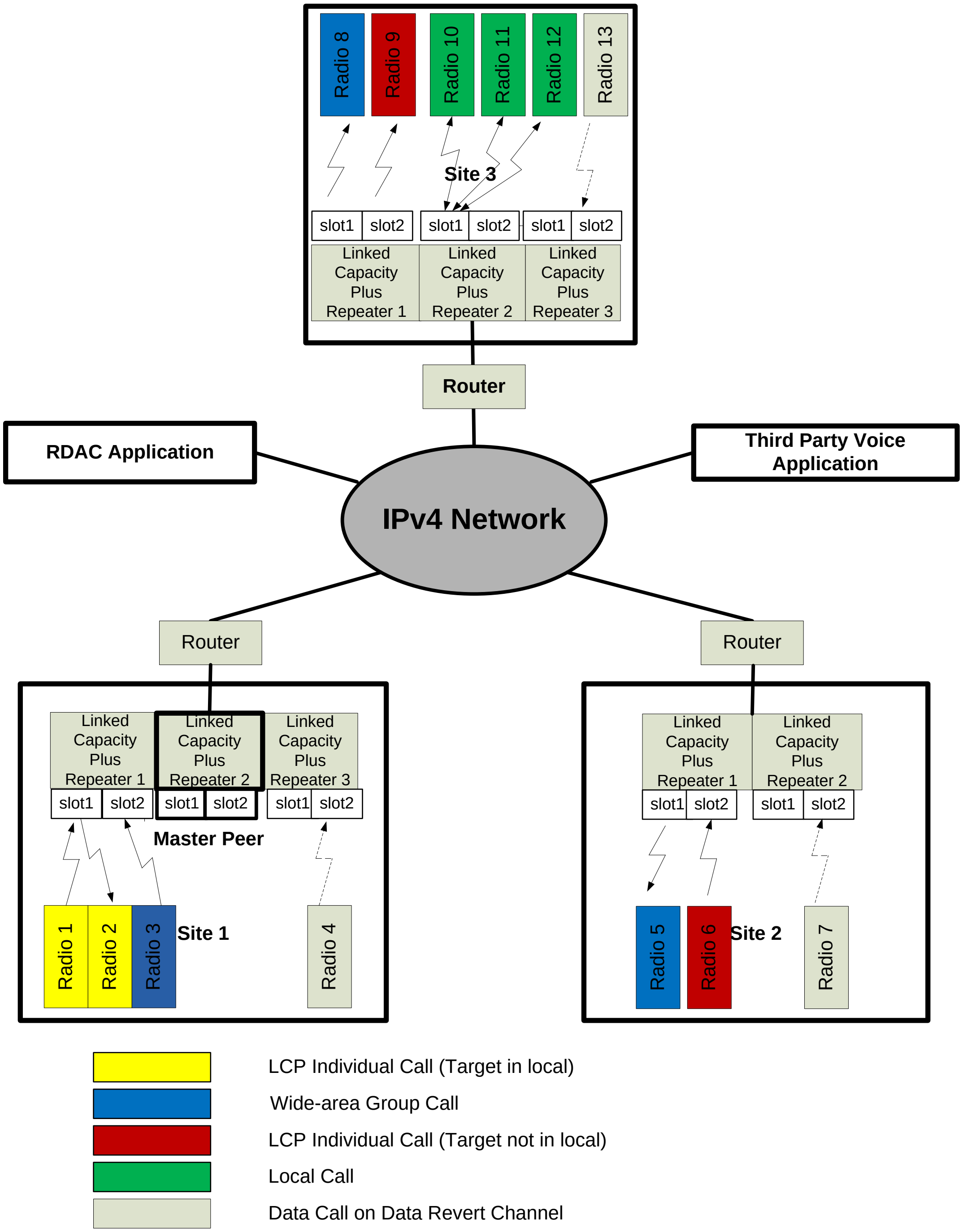


Figure 5: Linked Capacity Plus System with Third Party Application

Similar to the Site Peer concept in the Capacity Plus system, to make the application and the repeaters at different sites transparent to the Rest Channel movement in one site, the repeater peers in the Linked Capacity Plus system support IP aliasing by associating a static IP address and UDP port with the Rest Channel. This static IP address and UDP port seems to be another peer in the system. We call it the Site Peer.

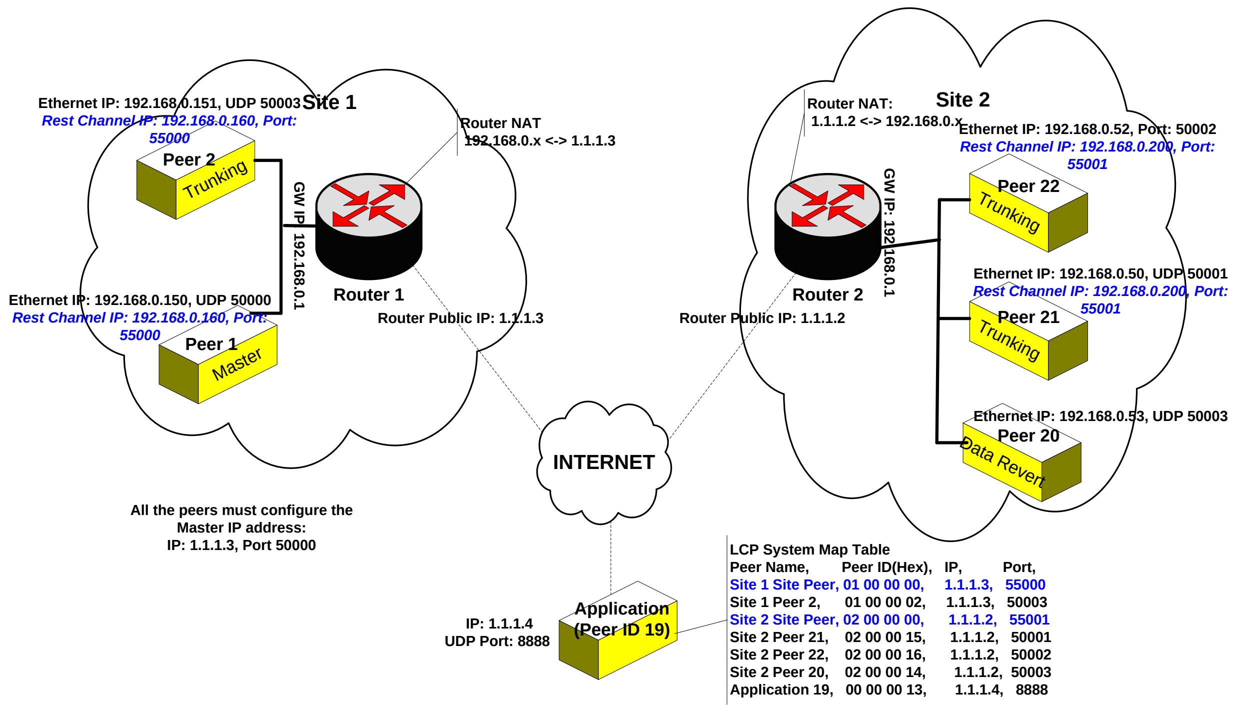


Figure 6: Example LCP IP Schema and System Map

Each site has a Site Peer with a unique IP address/UDP port, which any repeater at other site or third party application can use to send a message. In Figure 6, site1's Site Peer has the IP address of 1.1.1.3, UDP port of 55000; and site2's Site Peer has the IP address of 1.1.1.1 and UDP port of 55001.

When the switch at the site receives the message targeted to the Site Peer's IP address/UDP port, it sends the message to the repeater that last used the Site Peer's IP address/UDP port. The repeater changes the destination address to the IP Limited Broadcast address and transmits over the LAN so that all the repeaters receive the message at the same time. The broadcast messages may have some adverse effects on the other devices present on the LAN. Therefore it is recommended that only the LCP repeaters are present on the LAN. Third party applications shall resident on a different LAN from the LCP repeater peers. From the repeater peers point view, the third party application is another Site Peer with Site ID of 0.

The Rest Channel Repeater at a site is the Site Peer, it has the following responsibilities:

- Registers with the Master Peer if the Site Peer for this site is not registered yet.
- Sends Keep Alive messages periodically to all the other Site Peers and applications using the Site IP addresses in the source and destination address. This keeps the firewall at all the sites open for the inter-site communication.

At each site, all the repeaters are behind a router that supports "Basic NAT". As seen from the WAN, the repeater LAN is behind a single IP address, the Site Peer IP address. But each of repeaters has a unique UDP port. The Site Peer IP address together with the unique UDP port is called the individual repeater WAN IP address. As shown in the system map shown in Figure 6, all the repeaters from the site2 has the WAN IP address of 1.1.1.2, but the UDP port for each of them are different.

Each repeater peer registers with the Master Peer in the system, the Master Peer distribute the individual repeater WAN IP address to all the peers in the system. Also each repeater peer sends periodic Keep Alive messages to the Master Peer to keep the firewall open and minimize the probability of its UDP port getting de-allocated.

The Repeater Site Peer IP address and port is used only during the call set-up. Once the participating repeaters exchange their own individual WAN IP addresses, the remaining call messages must use the participating repeater's individual WAN IP address.

Unlike repeater peers in the IPSC or Capacity plus system, the Peer ID of the repeater peers in the LCP consists of a site ID and a repeater ID (Radio ID field in the MOTOTRBO CPS). All the repeaters in the same site have the same site ID, which is the "Site ID" in the MOTOTRBO CPS. The Site Peer's repeater ID is always 0. The repeater ID shall be unique across the LCP system otherwise the wide area group call may not be received in all the sites. For example, it is not valid to have repeater ID of 1 in both site1 and site2. The third party application peers' Site ID has to be 0, otherwise the repeater does not accept the wireline registration request. See Figure 6 for the example system map which includes the third party application, the repeater peers and the Site Peers.

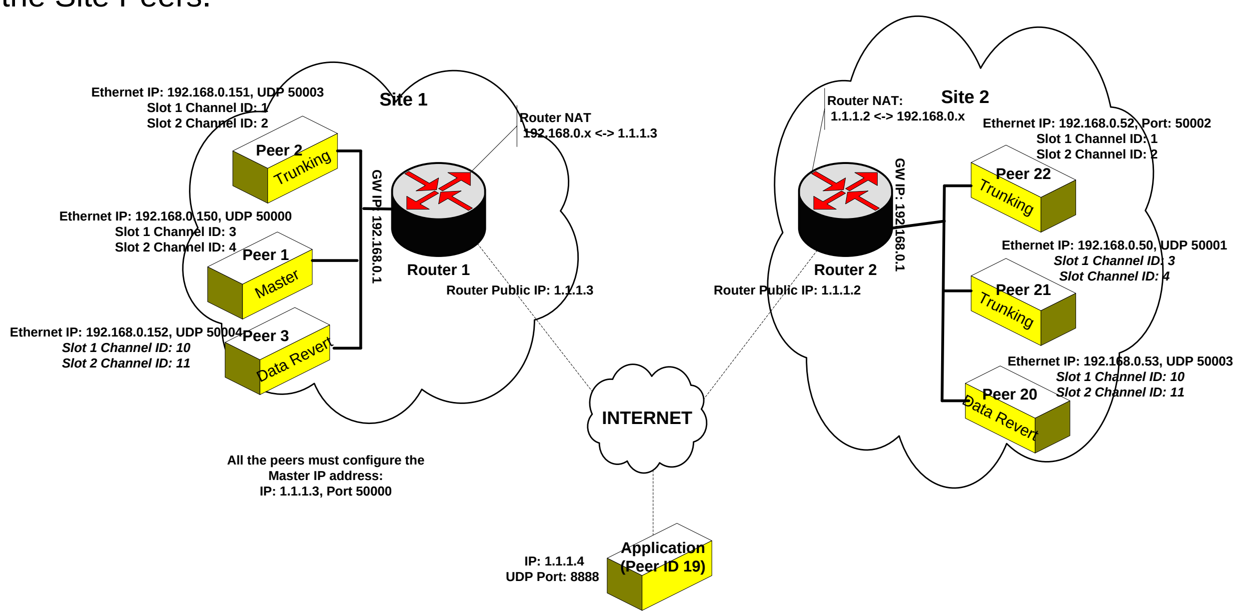


Figure 7: Example LCP Channel ID Assignment

Each trunking repeater has two logical channels. From the MOTOTRBO CPS, the end user needs to assign unique Ids to all the channels at a site. If the Id of the first channel of a repeater is 'n' then the Id of the second channel is 'n+1'. The scope of channel ID is within a site. Therefore different sites can have the same channel IDs. As shown in Figure 7, within site1 or site2 all the peers have unique channel ID. But Peer1 at site1 can have the same channel ID as Peer21 at site 2.

To form a Wide Area Data Revert Channel across the data revert repeaters, the channel ID has to be the same as well as the slot number. In Figure 7, two Wide Area Data Revert Channels are formed: slot1 of peer3 at site1 and peer20 at site2; slot2 of peer3 at site1 and peer20 at sit 2.

2.5 *Call Control Interface Overview*

As illustrated in Figure 2, the IP Site Connect system uses IP version 4 (IPv4) based back-end network to connect the MOTOTRBO repeaters at different sites. The repeaters in the Capacity Plus system must be in the same LAN as illustrated in Figure 3. The IPv4 back-end network can be either a private IP network or the public internet network provided by an Internet Service Provider (ISP). The Linked Capacity Plus System has the same requirement in each site as the Capacity Plus System. It supports all the backend networks supported by the IP Site Connect except "dial-up" connection (due to small bandwidth) or Satellite Internet access (due to large delay). Third party applications can connect to the following systems through IPv4 based back-end network and become a peer:

- Single Site Conventional System
- IP Site Connect System
- Capacity Plus System
- Linked Capacity Plus System
- Analog Repeater System (Only for RDAC application)

Please note, in this document, the trunking system means Capacity Plus System and Linked Capacity Plus System; the conventional system means Single Site Conventional System and IP Site Connect System.

The messages defined in the Call Control interface follow the common message structure of the Network Application Interface: All the messages start with a 1-byte opcode and a 4-byte Peer ID. After that the message structure varies dependent on the sub-interfaces. For example, the Network Application Interface opcode for the Call Control Interface is 0xB2, for XCMP interface 0x70. We call the 1-byte opcode and 4-byte Peer ID the Network Application Interface Header.

In all the messages at the Call Control Interface, after the Network Application Interface Header there is a Call Control opcode field which defines the message types. See section 6.1 for the detailed message structure definition. Table 1 lists all the Call Control opcode assignments. The names of all the messages at the Call Control interface start with "WL", which stands for WireLine. This is used to differentiate the messages used in the pre-R2.2A IPSC protocol at the repeater interface. In this document, Wireline and Call Control have the same meaning.

Opcode	Value	Description	Related Call Type
WL_REGISTRATION_REQUEST	0x01	Third party application to subscribe calls of interest with repeater peers	Voice, CSBK
WL_REGISTRATION_GENERAL_OPS	0x03	De-register or Query existing registration entries	Voice, CSBK
WL_REGISTRATION_STATUS	0x02	Response to Wireline Registration Request and Wireline Registration General Operations	Voice, CSBK
WL_CHNL_STATUS	0x11	Report channel status to the third party application. For trunking system, this message reports the Rest Channel status	Voice, CSBK
WL_CHNL_STATUS_QUERY	0x12	Third party application to query the channel status. The repeater peer responds with WL_CHNL_STATUS	Voice, CSBK
WL_VC_CHNL_CTRL_REQUEST	0x13	Third party application to initiate a voice or CSBK call	Voice, CSBK
WL_VC_CHNL_CTRL_STATUS	0x16	Response to Wireline Channel Control Request.	Voice, CSBK
WL_VC_CSBK_CALL	0x17	Receive CSBK call request or CSBK call response from the radio	CSBK
WL_VC_VOICE_START	0x18	Indicate the beginning of a voice call with call attributes	Voice
WL_VC_VOICE_BURST	0x21	Each message carries 60ms the audio data	Voice
WL_VC_VOICE_END	0x19	Indicate the end of a voice call	Voice
WL_VC_VOICE_PRIVACY_BURST	0x22	Specify privacy information when enhanced encryption is enabled	Voice
WL_VC_CALL_SESSION_STATUS	0x20	Indicate call session status	Voice, CSBK

Table 1: Call Control Opcode Assignment

Before a third party application can use the Call Control interface to initiate or receive a call, it has to finish the following two steps:

- Following the Link Establishment protocol to join the repeater system and maintain the links with other peers in the system, See Section 3.1 for the specific setting needed for the third party application information.
- Finish the Call Control Interface's Wireline Registration with each of the repeater peers to register the application's interested call type and range. See Section 3.2 for more detailed information.

When a MOTOTRBO repeater peer receives a voice call over the air, it creates the WL_VC_VOICE_START message by extracting the call information from the Digital Mobile Radio (DMR) voice header burst, and sends the message to the registered third party application. Then at 60ms interval, it sends the WL_VC_VOICE_BURST message to the third party application. The WL_VC_VOICE_BURST message contains a Real-time Transport Protocol (RTP) header for the 60ms audio data. The RTP protocol is for the media stream, e.g. voice, to allow synchronization and jitter calculations and identify the type of carried content. When the voice stream ends, the repeater sends WL_VC_VOICE_END message to indicate the end of the call session, and WL_VC_CALL_SESSION_STATUS message with the status of call hang so that other parties can talk back. See section 3.5 for more detailed description on the voice call.

Figure 8 shows the protocol stack for the voice call set-up and call session status.

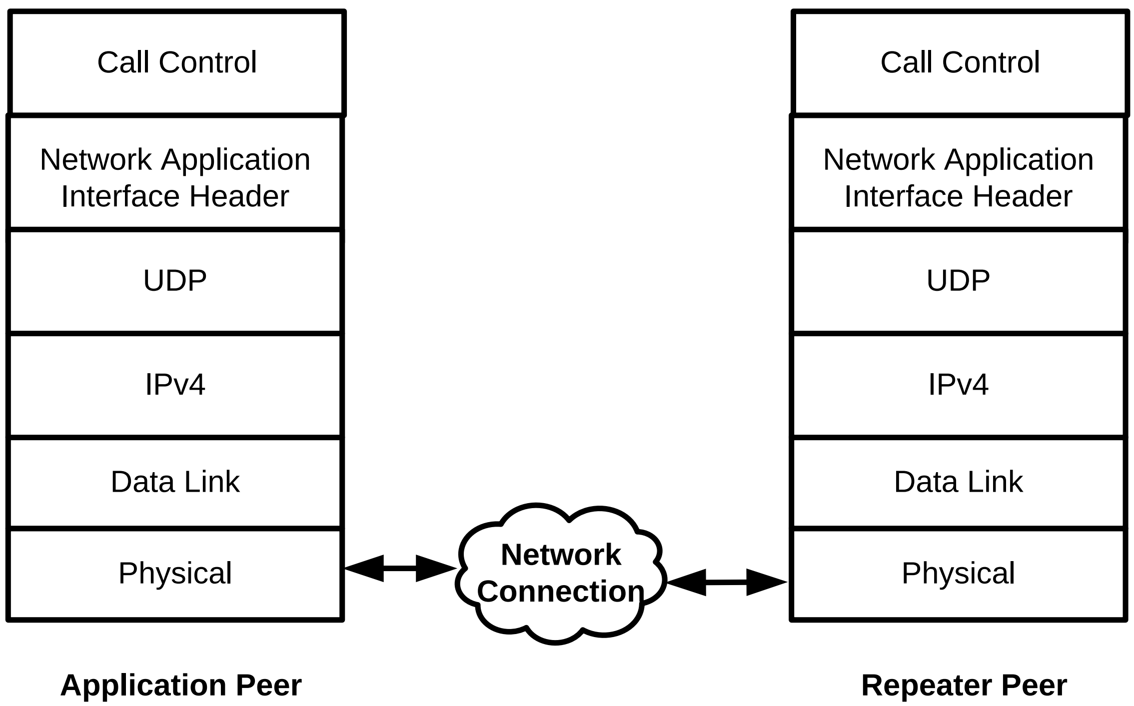


Figure 8: Protocol Stack for Call Control Interface – Voice No Audio Stream Part / CSBK Call

Figure 9 shows the protocol stack for the voice call audio streaming.

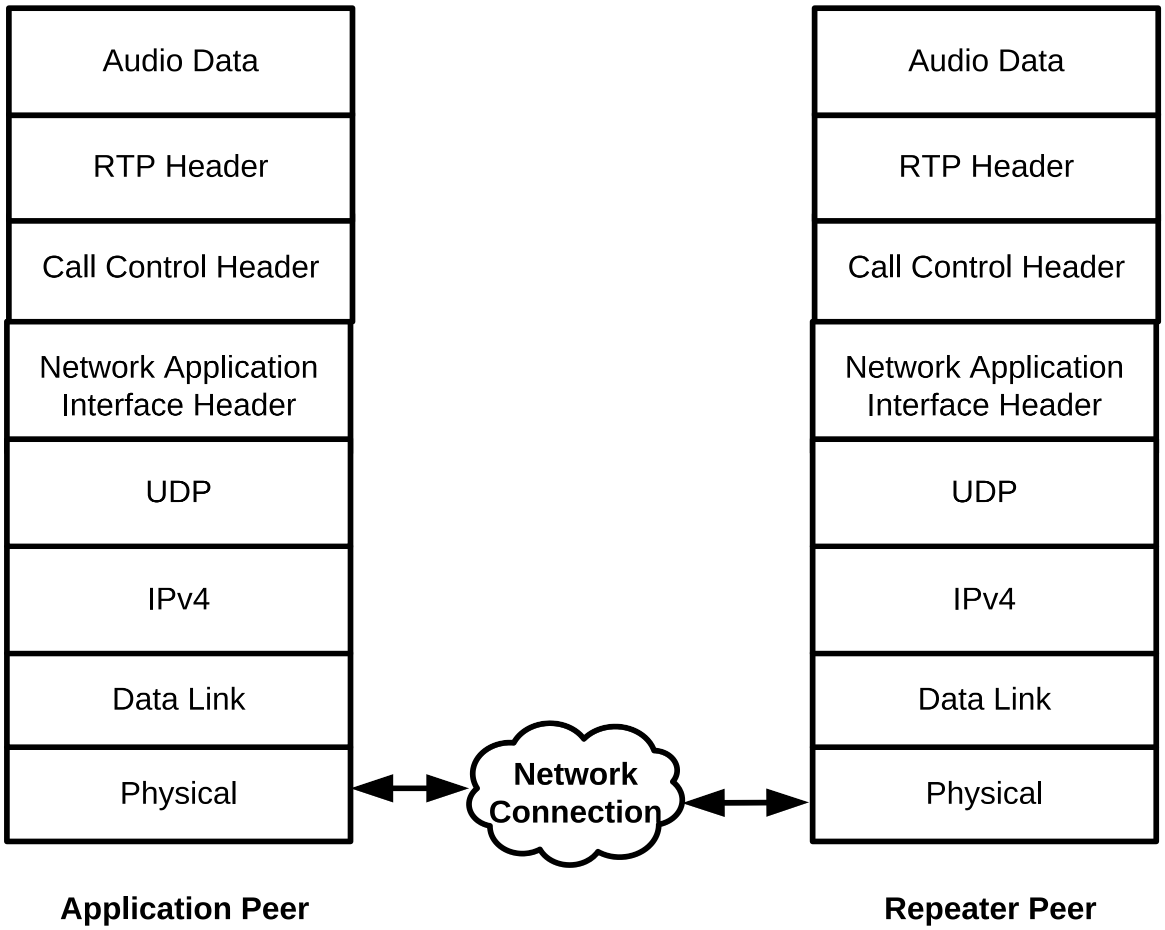


Figure 9: Protocol Stack for Call Control Interface – Audio Stream Part

When a MOTOTRBO repeater peer receives a CSBK call over the air, it generates the WL_VC_CSBK_CALL message by extracting the call information from the DMR CSBK burst to the registered third party application. The third party application uses the WL_VC_CHNL_CTRL_REQUEST to send the CSBK acknowledgement to the radio. The protocol stack used in the CSBK call is the same as shown in Figure 8. See section 3.6 for more detailed description on CSBK call.

2.5.1 Call Control Interface Capability

The Call Control interface is introduced since MOTOTRBO Repeater R2.2A. Therefore, the repeater software version has to be R2.2A or beyond and running on a repeater with 32MB memory. This requirement does not apply for the radio software version. Table 2 lists the system type and call type that supported in the Call Control interface. The cell that marked as NA means the call type is not supported by this system type.

Call Type	Call Direction	MOTOTRBOSystem Release					
		Single Site Conventional System	IP Site Connect System (Local or Wide Area Channel)	Capacity Plus System	Linked Capacity Plus System	Connect Plus System	Analog System
Voice Call	**From Radio to Third Party Application**	R2.2A	R2.2A	R2.2A	R2.2A	NA	NA
	From Third Party Application to Radio	R2.2A	R2.2A	R2.2A	R2.2A	NA	NA
CSBK Call	**From Radio to Third Party Application**	R2.2A	R2.2A	R2.2A	R2.2A	NA	NA
	From Third Party Application to Radio	R2.2A	R2.2A	R2.2A	R2.2A	NA	NA

Table 2: System Release that Enables the Call Type

The voice call in Table 2 includes the following call types:

- Confirmed Individual Call (Off Air Call Setup – OACSU)
- Unconfirmed Individual Call
- Group Call
- Emergency Call
- All Call

The following services for the voice call can be supported:

- Basic Privacy
- Enhanced Privacy

- Transmit Interruptible Voice Call Initiated by the Radio

The CSBK call in Table 2 includes the following call types:

- Call Alert / ACK
- Radio Check / ACK
- Emergency Alarm / ACK
- Radio Disable / ACK
- Radio Enable / ACK
- Remote Monitor / ACK
- IP Console Radio Inhibit / ACK
- IP Console Radio Un-inhibit / ACK

2.5.2 Interface Comparison

The voice/CSBK calls listed in Table 2 are also supported at the control station interface. There are advantages and disadvantages in the control station interface comparing with the Call Control interface. Table 3 summarizes the difference from the areas of required equipment, required software, voice/data/control call processing. The big advantages on the connection through Call Control interface are in the areas of required equipment, required software, digital audio access, and location to run the server application. The control station based deployment for IP Site Connect or Single Site system requires Multi-Channel Device Driver (MCDD) for the outbound message routing. However MCDD can only support up to 16 control stations. Therefore the maximum number of supported IP Site Connect systems with 2 WAC is 8. The Call Control interface based deployment does not need MCDD. It is up to the third party application how many repeater systems it can support.

The following items shall be kept in mind when connecting to the repeater systems through the Call Control Interface:

- Data call is not supported. The third party application shall use the MOTOTRBO Network Interface Service (MNIS) to send/receive data call.

- Num of supported site is reduced in IPSC and LCP system. There is a limitation of 15 maximum sites in a repeater system because each repeater can only support streaming two voice calls (one for each slot) to 14 other sites. Each application which receives audio stream from the repeater consumes one count. For example, if there are two voice applications connecting to an IP Site Connect system, maximum 13 repeater peers are allowed in the repeater system. If there are 3 voice applications connecting to a Linked Capacity Plus system, each application is counted as a Site Peer, therefore maximum 12 repeater sites are allowed.

- Max 4 voice applications can connect to one repeater system. It is recommended that if possible, to have one server application to access the repeater system and make the client applications connect to the server.

- The third party application has to conduct AMBE 2+ voice encoding / decoding.

	Connection through Control Station	Connection through Repeater Call Control Interface
Required Equipment	• Control station per each channel • Universal Serial Bus (USB) cable per each connection	• Repeater with 32MB memory • Ethernet cable • Router for a public IP network connection • Router or switch for a private IP network connection • Off-the-shelf low end router is sufficient
Required Software	• Multi-Channel Device Driver (MCDD) • MOTOTRBO Radio USB Driver	• None
Where a third party Server Application Has to Run	• At the PC or third party network device connecting to the control station	• At any PC or third party network device connecting to the IPV4 network
Impact to Max Number of Repeater Peers	• None	• Each voice application reduces the number of repeater peer by 1.
Max Supported Console Application	• No limite	• Limite 4 voice applications to join one repeater system
System Controller Mode	• Not support	• Not support
Voice Message Processing	• Decoded analog audio signal is available at the radio connector	• Simplified DMR Voice Burst assemble/de-assemble is required • AMBE 2+ encoding/decoding is required • Digital audio signal is accessible • Call Control interface protocol processing
Data Message Processing	• UDP/IPv4 message encapsulation/de-capsulation	• Not support
Control Message Processing	• XCMP message	• Simplified DMR CSBK Burst assembly/de-assembly • Call Control interface protocol processing

Table 3: Comparison between Call Control Interface and Control Station Interface

2.6 *Interface to Device Detection and Mobility Management (DDMS)*

2.6.1 Route Information

Before MOTOTRBO 2.2A system release, the Presence Server, which can be either a third party application or MOTOTRBO Presence Notifier (PN), tracks the radio presence based on the Automatic Registration Service (ARS) message, and notifies the third party application on the radio present status change. Starting from R2.2A, the MOTOTRBO PN is replaced with the software called Device Discovery and Mobility Service (DDMS) which has both the presence information and the route information. In this document radio route information has the same meaning as radio mobility information. Table 4 shows an example of mobility/route information on different system modes. NA means Not Applicable.

Mobility/Route Information	MOTOTRBO System Configuration				Notes
	Single Site Conventional System	IP Site Connect System	Capacity Plus System	Linked Capacity Plus System	
System Type	X	X	X	X	Always be MOTOTRBO
System Configuration	X	X	X	X	Specify the system operating mode
Link Establishment Domain	X	X	X	X	All the peers in a repeater system use the LE protocol to connect each other and form a LE domain. This field specifies the Master Peer IP address and UDP Port at the LE domain the repeater system belongs to.
Site ID	NA	NA	X	X	The Capacity Plus system can be viewed as a single site Linked Capacity Plus system, its Site ID is always 0.
Slot Type	Local	Wide/Local	NA	NA	Specify the slot type
Slot Number	X	X	NA	NA	Specify if it is slot 1 or slot 2

Table 4: Radio Mobility Information

Let us start with a simple example. When a third party application initiates a call to a remote radio in a Single Site system, if it knows which slot the target radio exists based on the previous inbound message, the third party application can directly place the call on that slot instead of placing the call on both slot1 and slot2.

Another example is a third party application connects to three IP Site Connect systems and one Single Site system at the same time. The radio mobility information in the DDMS tells the target radio is at which LE domain, and at which slot. The third party application can use this information to place the call exactly at that LE domain and at that slot. Initiating calls based on mobility information significantly optimizes the RF channel usage.

The third example is a third party application connecting to a 15-site Linked Capacity Plus system. When initiating an individual call to a remote radio, since the third party application knows which site the radio is, it can directly send the call request to the Site Peer at that site with the option of “Not Forward to Remote Site” so that the unnecessary busying of the remote sites is avoided and call setup delay is reduced.

2.6.2 Interface to DDMS

Figure 10 shows all the interfaces used by the third party voice application as well as the MNIS and third party data application. Here we only focus on the DDMS watcher interface. For detailed explanation on MNIS and third party data applications, please refer to MOTOTRBO Network Interface Service ADK Development Guide.

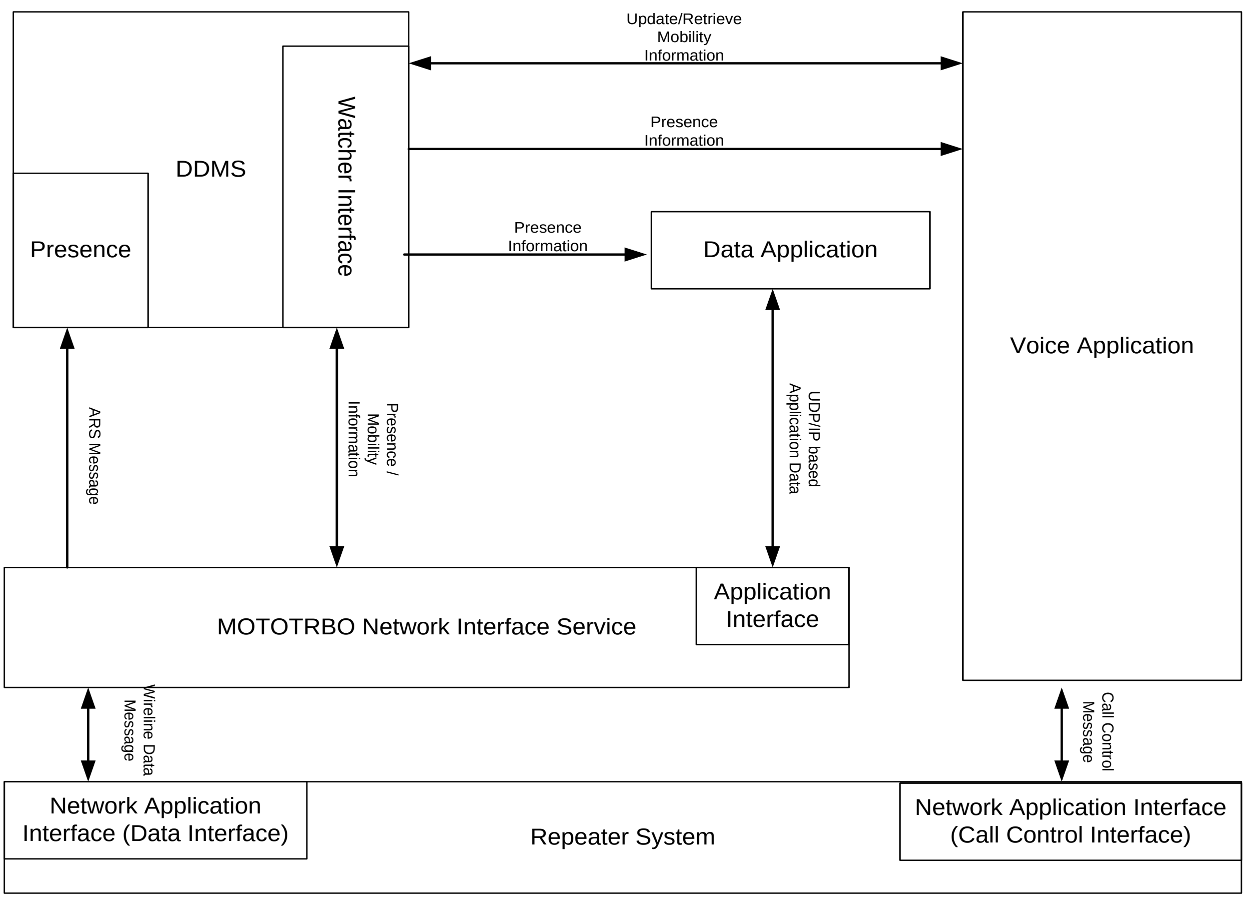


Figure 10: Interface Overview

The DDMS watcher interface allows applications (including MNIS) to update/retrieve radio presence and route information from the DDMS.

The DDMS maintains both the radio presence and route information. It provides interface to the MNIS and the third party voice or data application to get notification on the presence change, and also to the MNIS and the third party voice application to get and update route information.

- Presence Service: The MNIS forwards the radio ARS message to the DDMS which updates the radios presence. The DDMS notifies applications that have subscribed for presence through the Watcher interface.
- Route Information: Either the MNIS or third party voice application uses the Watcher interface to update the route information in DDMS. When the mobility information is updated both the MNIS and third party voice application get the notification. The MNIS updates the route information only derived from the received ARS message. In the trunking systems (Capacity Plus or Linked Capacity Plus system) when the ARS registration message is received at the data revert channel, the mobility information from this message can only tell the Site and LE domain. This is sufficient for the trunking system because all the idle radios follow the rest channel and the application always initiates the call at the rest channel.

Please note the DDMS only updates presence based on ARS messages. MNIS shall be installed for the DDMS to receive the ARS message as shown in Figure 11. ARS feature has to be turned on in the radio for the DDMS to track the presence of the radio. The DDMS is required to be always on to acknowledge the periodic ARS registration message from the radio. Otherwise, the channel is heavily occupied by the ARS registration re-try messages.

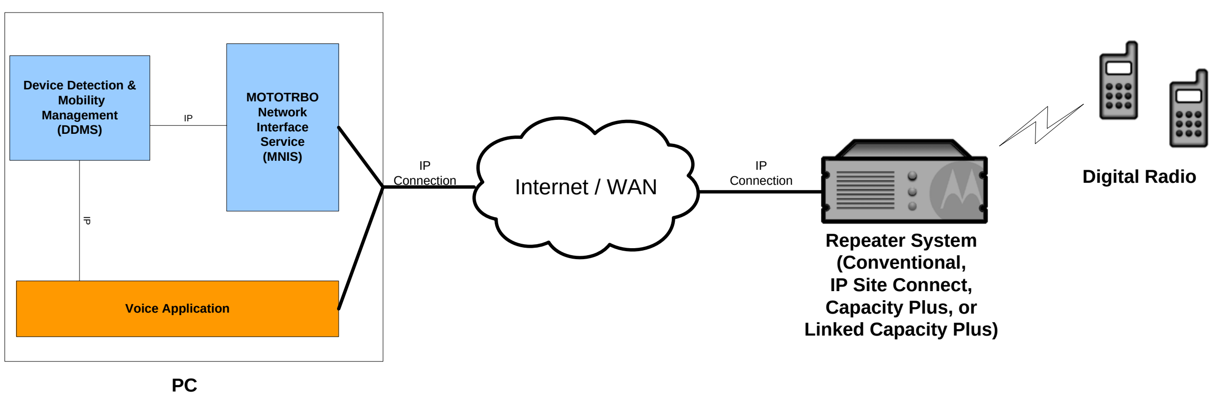


Figure 11: Voice Application with DDMS

The Watcher interface shown in Figure 10 is similar to the original PN Watcher interface with the following difference:

- TCP connection must be used
- DDMS is the TCP Server, the Watchers are the TCP clients
- Use the new SUBSCRIBER V2 message for subscription
- Only Reliable Notification message is supported over the TCP connection
- New messages are added to support mobility update

Refer to [6] MOTOTRBO Device Discovery and Mobility Management-to-Watcher Interface Protocol Specification for further information.

2.7 *Example System Topologies*

2.7.1 Voice and Data Application Server

The third party application uses the Call Control Interface to send/receive voice/CSBK call to/from the MOTOTRBO radio systems. The third party application uses the MOTOTRBO Network Interface Service (MNIS) interface to send/receive data call to/from the MOTOTRBO radio systems. Figure 12 shows both the third party data application, and the third party voice application are in the same PC as the MNIS and DDMS. Since the communication paths among MNIS, data application, DDMS and voice application are IP based, there is no restriction where those software components shall resident as long as the IP routing path is set up correctly.

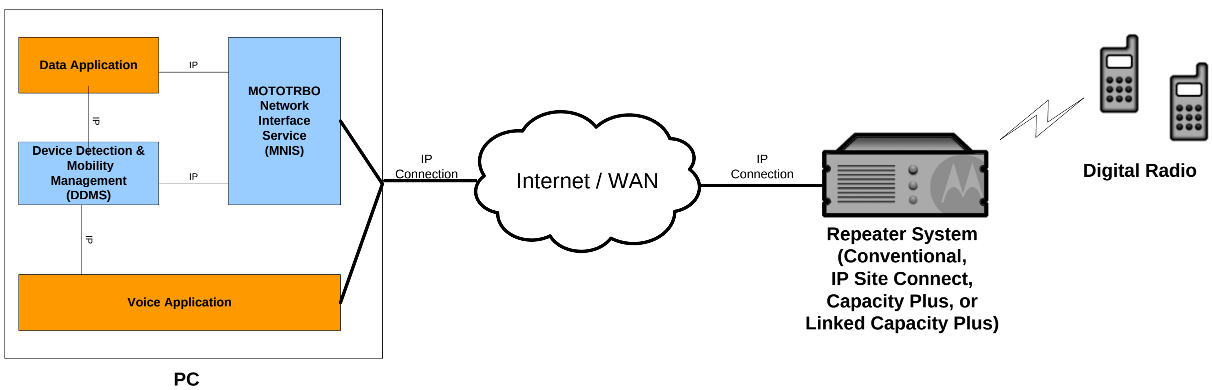


Figure 12: Centralized Voice and Data Application Server

2.7.2 Multiple IP Site Connect Systems with Centralized Application Server

The co-existence of multiple IP Site Connect systems at the same site can be the solution to increase the wide area capacity for a customer. For example, both site1 and site2 have 4 WACs available, two from the IP Site Connect system1 and two from the IP Site Connect system2. Each IP Site Connect system could have its own distributed third party application.

When there is a need to have a centralized application server for both IP Site Connect system1 and system2, there are two independent ways to accomplish it:

- Radio Control Station based, which is shown in Figure 13
- Call Control Interface based, which is shown in Figure 14.

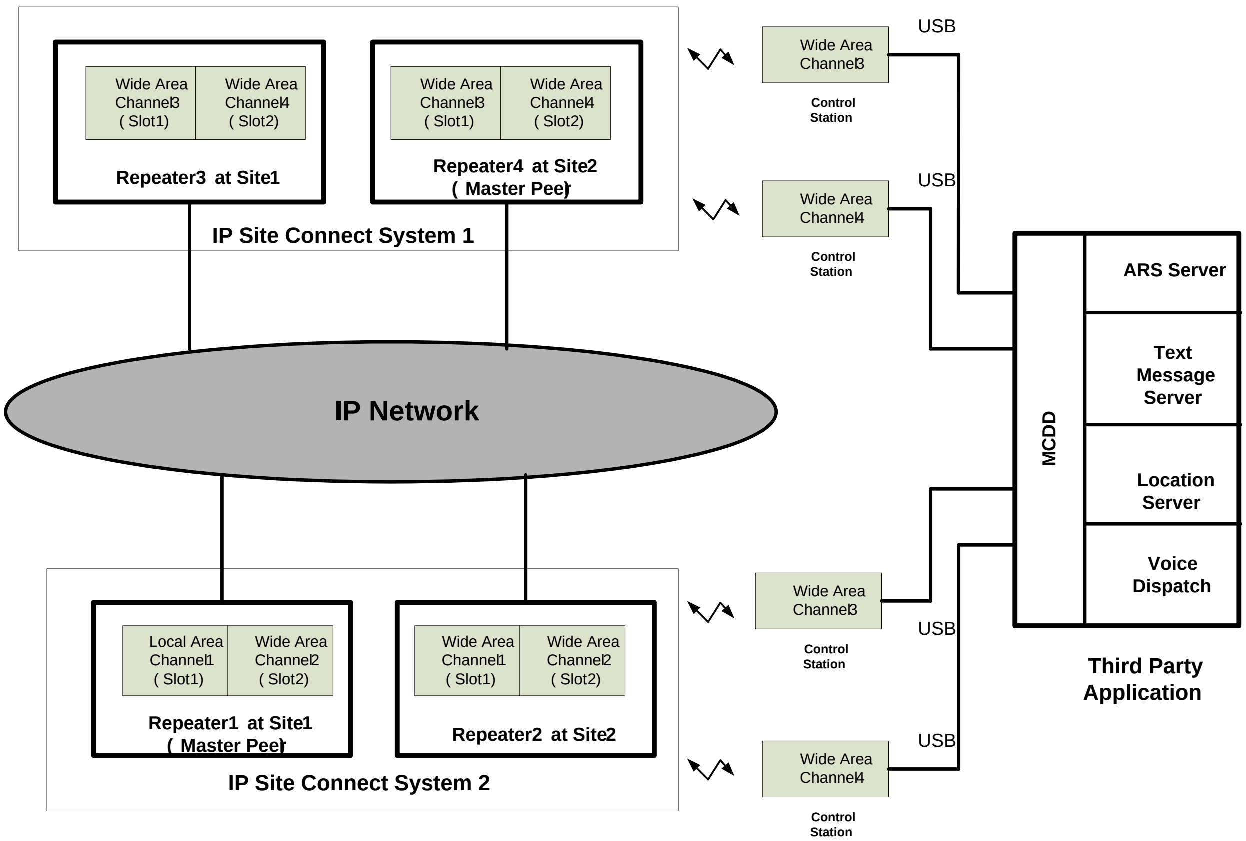


Figure 13: Multiple Wide Area System with Centralized Application Server through Radio Interface

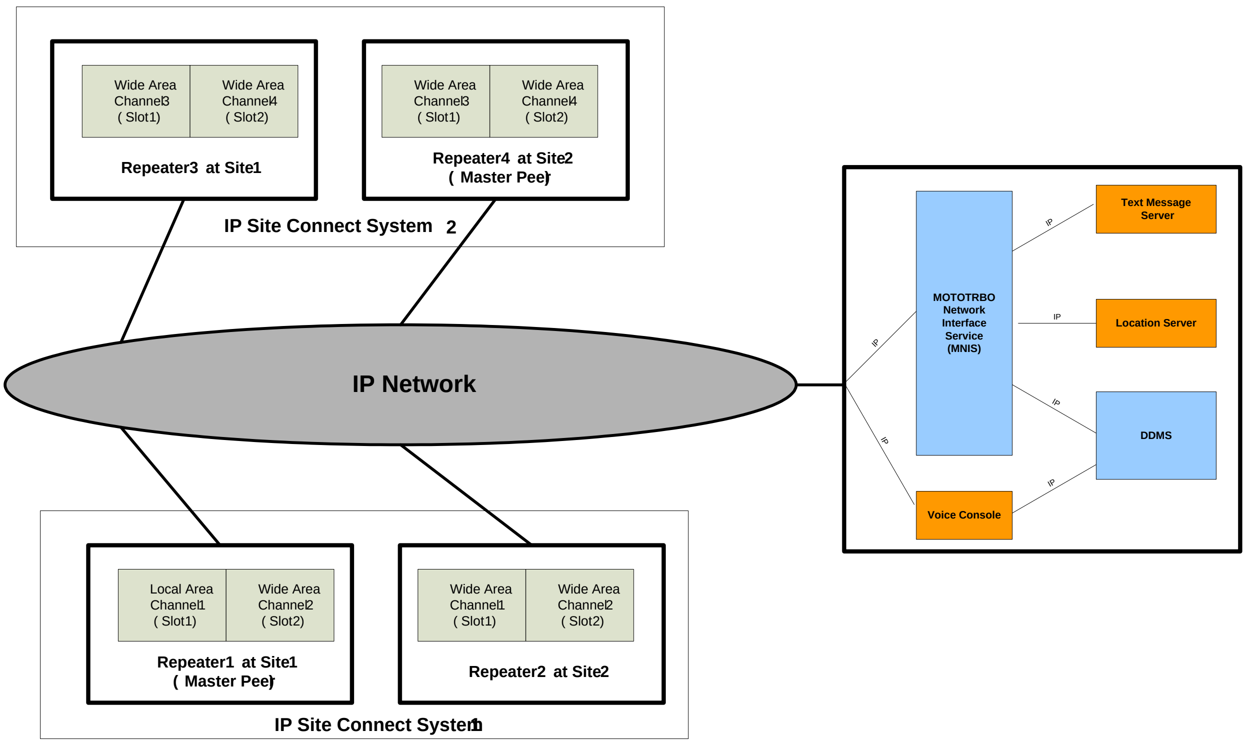


Figure 14: Multiple Wide Area System with Centralized Data Application Server

There are advantages and disadvantages in both ways of connection. Table 3 summarizes the difference from the areas of required equipment, required software, voice/data/control call processing. The big advantages on the connection through the Call Control interface are in the area of required equipment, digital audio access, and location to run the server application.

2.7.3 Wide and Local Area IP Site Connect Systems with Centralized Application Server

In case a customer has a significant load of traffic in a certain site, it is possible to configure the repeater to have one LAC and one WAC so that the local traffic is separated from the wide area communication across the IP Site Connect system.

Figure 15 shows an example of such configuration in which slot2 is used for local communication at each site and slot1 is used for wide area communication. The centralized console application can send/receive voice call or CSBK calls on both WACs and LACs.

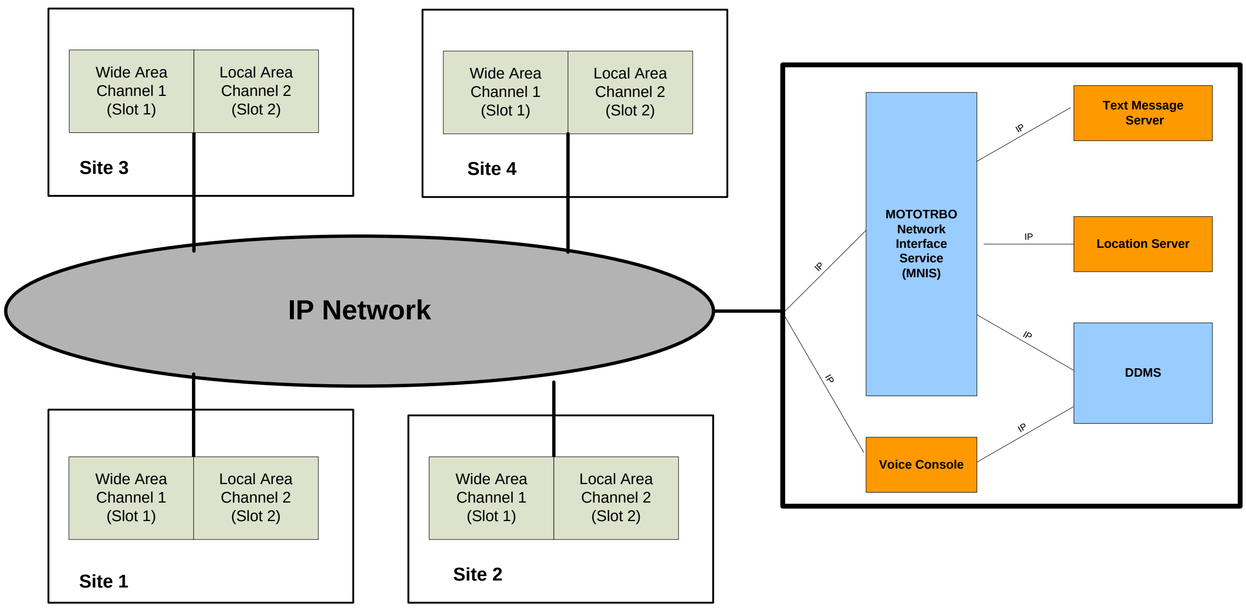


Figure 15: Wide and Local Area Systems with Centralized Data Application Servers

2.7.4 Centralized Application Server for Multiple Systems

As we discuss in Reference [1] LE ADK, it is possible for one third party application to join different systems at the same time. Figure 16 shows an example that a third party console application communicates with any of the radio for voice/CSBK calls in both Linked Capacity Plus and IP Site Connect system. The Call Control interface protocol has no restriction on how many repeater systems can be supported by the third party application.

The third party application joins each system by starting the LE registration process with the Master Peer in the systems. The system ID in the LE protocol version field has to be the system type the application intends to join in. The third party application conducts the Wireline Registration with each of non-data-revert repeater in the system. Because the same application can communicate with two repeater systems at the same time, unique radio ID shall be ensured across the systems.

Two instances of MNIS are installed in two PCs in Figure 16. This is due to the limitation of MNIS: one MNIS can only support one Linked Capacity Plus or Capacity Plus system, and two MNISs cannot co-exist in the same PC. However, one instance of DDMS can support both IP Site Connect and Linked Capacity Plus system at the same time.

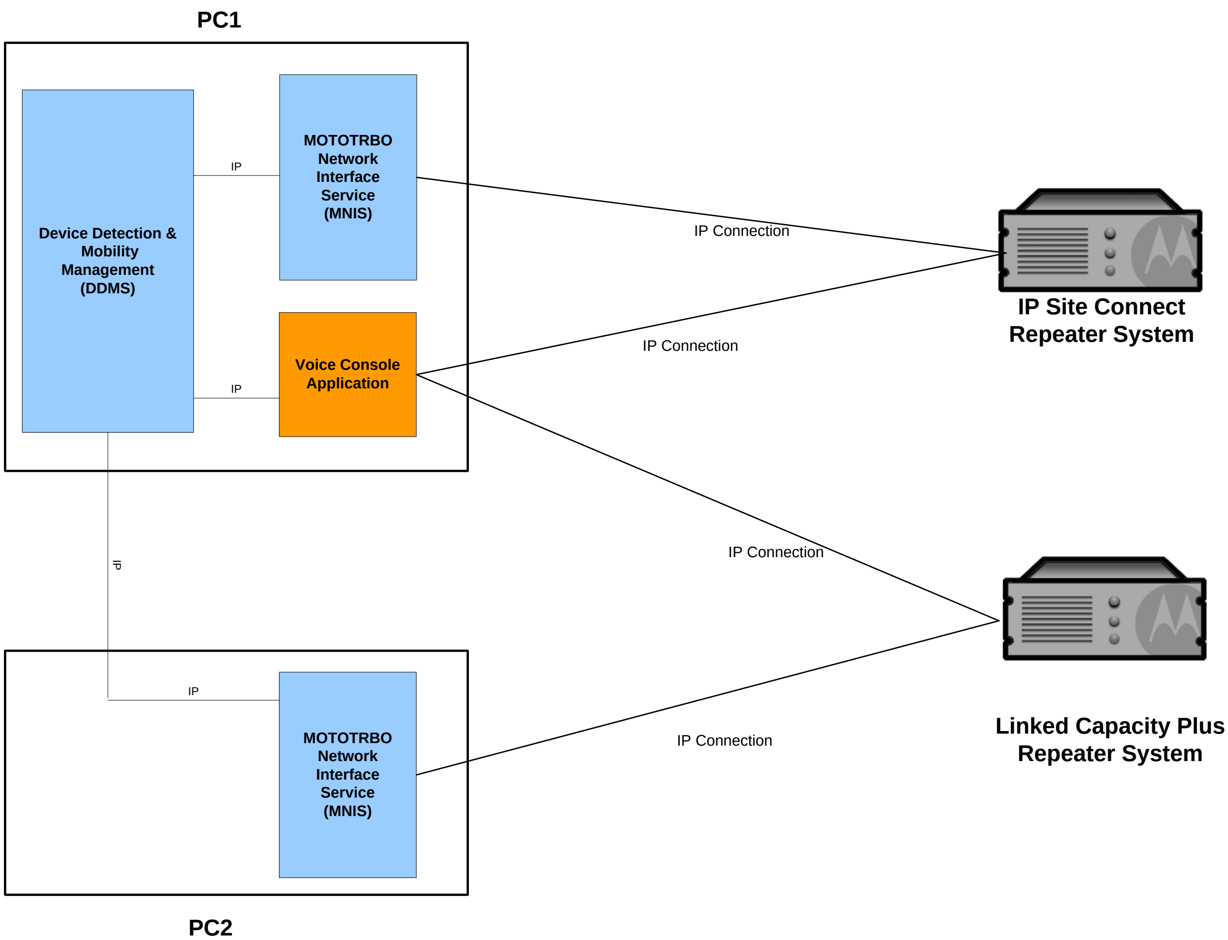


Figure 16: Centralized Third Party Application for Linked Capacity Plus and IP Site Connect System

3.0 Call Control Interface Protocol

3.1 Peer Discovery through Link Establishment

Before a third party application sends Call Control messages for voice/CSBK call, it has to join the repeater system as a peer using the LE protocol. After it sends out the Link Establishment (LE) registration message to the Master peer in the repeater system, the Master peer sends out the updated the system map once accepting the registration request

The third party application uses the LE_MASTER_PEER_REGISTRATION_REQUEST and LE_MASTER_PEER_KEEP_ALIVE_REQUEST to register service with Master Peer, and uses the LE_PEER_KEEP_ALIVE_REQUEST to register service bit with other peers.

The third party application needs to register and maintain the LE connection with all the repeater peers including the Site Peers, so that it can make Voice/CSBK call in the radio system. It also needs to register and maintain the LE link with all the RDAC peers, MOTOTRBO Network Interface Service (MNIS) peers to avoid the unnecessary peer re-discovery and system map broadcast messages. The requirement on the LE connections between the data revert repeater and the third party applications is dependent on the system mode.

- In the LCP system it is optional for the application to register with the data revert repeater peer. The data revert repeater peer can be identified in the Slot Assignment Bit of the peerMode field in the LE messages (For more details, refer to [1] Table 14). The application can set the “No Link Establishment with Data Revert Repeater” bit in the peerMode field so that the data revert repeater will not send keep alive messages to the application.

- In the CPC and IPSC system, the application shall connect to all data revert repeater peers. In the IP Site Connect system and single site conventional system, the application needs to track which slot of which repeater is used for GPS Revert channel to avoid initiating call at the GPS Revert Repeater peer. In the Capacity Plus system, the application uses the Site Peer to initiate a call, it does not need to track which repeater is used for the Data Revert Channels.

Below is an example LE registration procedure in the CPC system. Besides establishing links with all the repeater peers, the third party application has to link with the Site Peer, which has the peer ID of 0x00 in the LE system map. The same Rest Channel IP address and Port is configured in all the repeater peers in the system. Refer to Section 5.1 of this document for the Rest Channel IP address and Port configuration. The movement of the Rest Channel is transparent to the third party application. When the Rest Channel moves, the new Rest Channel repeater peer maintains the Site Peer link with the application. Between the third party application and all the repeater peers (including the Site Peer) have to maintain the LE connection by exchanging the periodic

Keep Alive messages. These Keep Alive messages keep the communication path between the applications and the peers open.

Note: Application can establish link with Rest Channel Repeater when it is available in the LE system map and is no need to wait all the repeater peers to join the Capacity Plus System. The third party application shall register the same service with all the repeater peers in the Capacity Plus system.

The LCP system has the same LE registration procedure as the CPC system except that the third party application has to register with all the repeaters and Site Peer in all the sites.

For detailed LE procedures in the conventional systems, refer to [1] section 3.2.1 Join Repeater System.

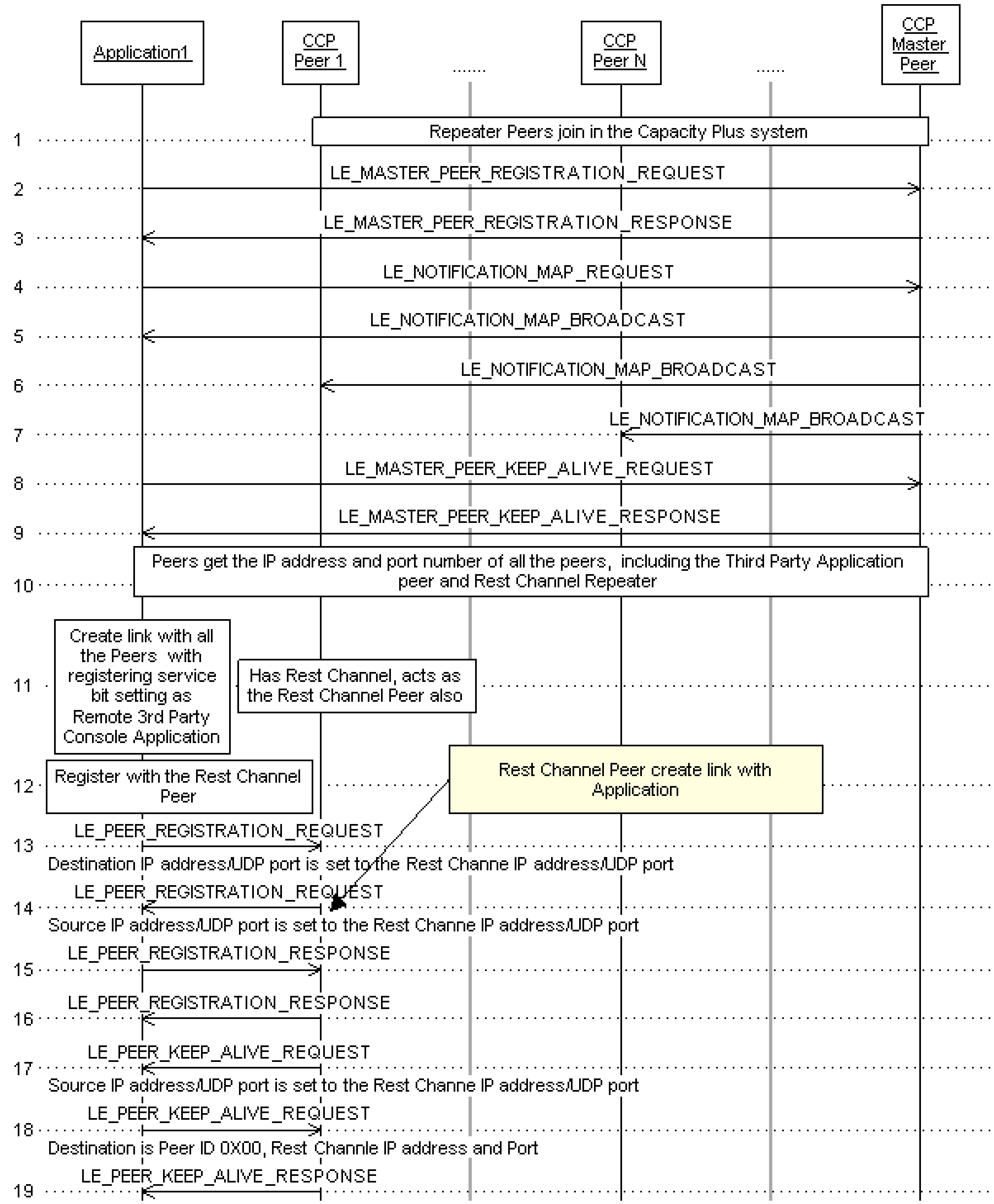


Figure 17: LE Registration in CPC System

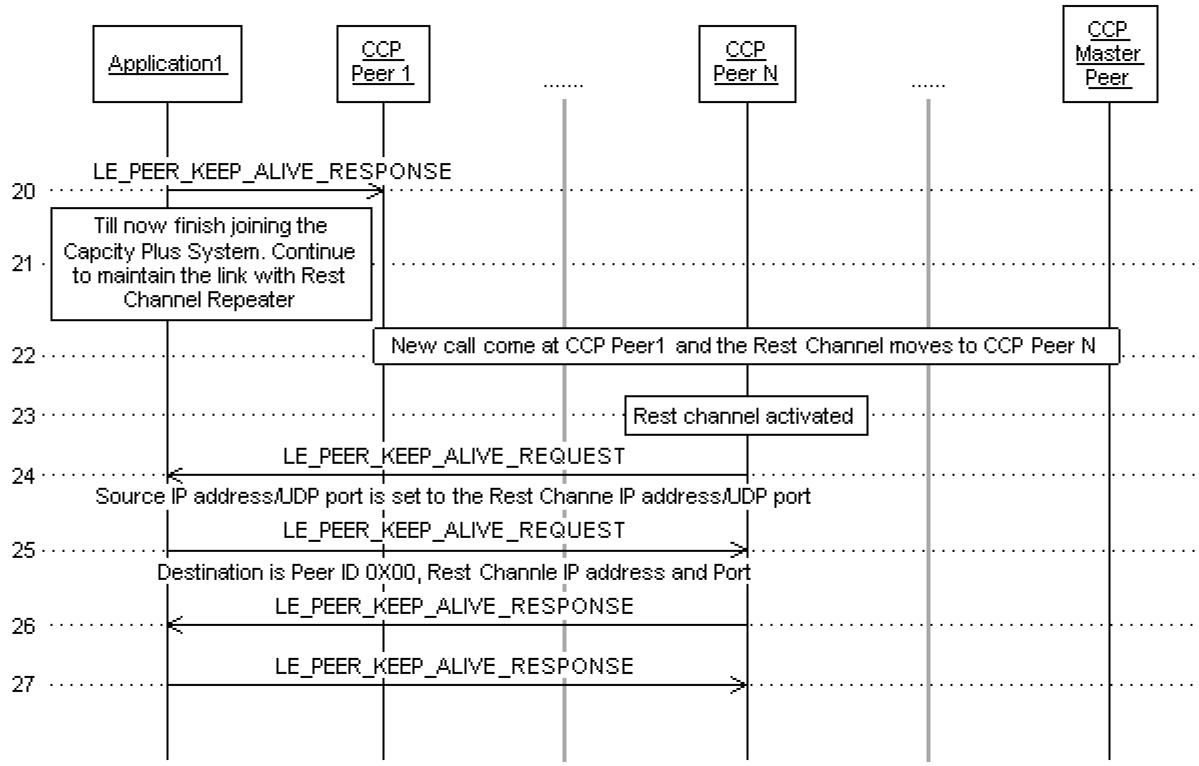


Figure 18: LE Registration in CPC System (Continue)

3.1.1 Setting in the LE messages

Below sections define the special bit settings in the LE message field to support the Call Control interface. For all other field and bit settings, see Reference [1].

3.1.1.1 Peer Mode Bit Field (peerServices) in LE protocol

The peerMode field specifies the peer's current operating mode. The tables below are the supplementary explanation for section 4.9.7 Peer Mode Bit Field (peerMode) in Reference [1].

In LCP system, the third party application must mark itself as the third party application peer in the peerMode field of its LE registration request message; the Site Peer and the MNIS Peer also set the corresponding bits to identify themelves in the peerMode field.

In Single Site, IPSC, CPC and LCP systems, the third party application shall mark itself as enabled in the Peer Status bit.

System	Version Introduced	
IP Site Connect	0	
Capacity Plus	0	
Peer Services Bit	**Peer Mode Name**	**Description**
0-5	Refer to [1] section 4.9.7	
6-7	Peer Status	Indicate if a peer is currently disabled or enabled. 0b01: Yes 0b00: No

Table 5: Peer Mode Bit Allocation in IPSC/CPC

System	Version Introduced	
Linked Capacity Plus	3	
Peer Services Bit	**Peer Mode Name**	**Description**
0-3	Refer to [1] section 4.9.7	
4-5	Peer Status	Indicate if a peer is currently disabled or enabled. 0b01: Yes 0b00: No
6	Refer to [1] section 4.9.7	
7	Remote 3rd Party Application	Indicates if the peer is a remote 3rd party application peer: 1: Yes 0: No
8-9	Refer to [1] section 4.9.7	
10	LCP Site Peer	This bit indicates that the peer is a Site Peer. 1: Yes 0: No
11-13	Refer to [1] section 4.9.7	
14	MOTOTRBO Network Interface Service	This bit indicates if the peer is MNIS. 1: Yes 0: No
15	Refer to [1] section 4.9.7	

Table 6: Peer Mode Bit Allocation in LCP

3.1.1.2 Peer Services Bit Field (peerServices) in LE protocol

The peerServices field specifies the services supported by the associated peer. The tables below are the supplementary explanation for Reference [1] section 4.9.8 Peer Services Bit Field (peerServices). When a bit is enabled or set to 1, the associated service is supported by the peer. Otherwise, the service is not supported by the peer.

In the Single Site, IPSC and CPC system, the application must mark itself as the third party application peer in the peerServices field of its LE registration request message; the CPC Site Peer and the MNIS Peer also set the corresponding bits to identify themselves in the peerServices field.

System	Version Introduced	
IP Site Connect	0	
Capacity Plus	0	
Peer Services Bit	**Peer Service Name**	**Description**
The following 16 bits are allocated or reserved for support by all releases		

0-12	Refer to [1] section 4.9.8	
13	Remote 3rd Party Application	This bit indicates that the peer is a remote 3rd party application peer.
14-17	Refer to [1] section 4.9.8	
18	CPC Site Peer	This bit indicates if the peer is a Site Peer in a Capacity Plus system: 1: Yes 0: No
19-21	Refer to [1] section 4.9.8	
23	MOTOTRBO Network Interface Service	This bit indicates if the peer is MNIS 1: Yes 0: No
24-25	Refer to [1] section 4.9.8	
26	Network Application Interface Voice CFS enabled	This bit indicates that the repeater peer supports the NAI Voice/CSBK services based on Charge For Software (CFS) enablement, which is the Call Control interface enablement. 1: Yes 0: No
27	Wireline Service Enabled (slot 1)	This bit is set by the repeater. When set it indicates to the application that the repeater has its slot 1 wireline registration. 1: Yes 0: No
28	Wireline Service Enabled (slot 2)	This bit is set by the repeater. When set it indicates to the application that the repeater has its slot 2 wireline registration. 1: Yes 0: No
29-31	Refer to [1] section 4.9.8	

Table 7: Peer Services Bit Allocation in IPSC/CPC

System	Version Introduced	
Linked Capacity Plus	3	
Peer Services Bit	**Peer Service Name**	**Description**
0-10	Refer to [1] section 4.9.8	
11	Network Application Interface Voice CFS enabled	This bit indicates that the repeater peer supports the NAI Voice/CSBK services based on Charge For Software (CFS) enablement, which is the Call Control interface enablement. 1: Yes 0: No
12	Wireline Service Enabled (slot 1)	This bit is set by the repeater. When set it indicates to the application that the repeater has its slot 1 wireline registration. 1: Yes 0: No
13	Wireline Service Enabled (slot 2)	This bit is set by the repeater. When set it indicates to the application that the repeater has its slot 2 wireline registration. 1: Yes 0: No
13-31	Refer to [1] section 4.9.8	

Table 8: Peer Services Bit Allocation in LCP

3.1.1.3 Peer Services/Peer Mode Bit Field Usage in LE protocol

As indicated in section 3.1.1.1and 3.1.1.2 there are some of the bits in the peerService field and peerMode field the third party application peer must set. Many other bits in the peerServices field and peerMode field are intended only for the repeater peers. The third party application shall not set those bits. And some of the bits are optional for both the third party application and repeater peers. The table below provides the detailed bit settings in the peerService field and peerMode field per the peer type and per system type.

Peer Service Name	MOTOTRBO Repeater Peer		3rd party Voice/CSBK console	
	IPSC	CPC	IPSC	CPC
Master Peer	Optional	Optional	Optional	NO
Packet Authentication	Optional	Optional	Optional	Optional
Slot 1 Assignment in Capacity Plus	No	Yes	No	No
Slot 2 Assignment in Capacity Plus	No	Yes	No	No
Remote 3rd Party Application	No	No	Yes	Yes
Repeater Call Monitoring	No	No	Optional	Optional
Slot 1 Phone Gateway	Optional	Optional	No	No
Slot 2 Phone Gateway	Optional	Optional	No	No

Peer Service Name	MOTOTRBO Repeater Peer		3rd party Voice/CSBK console	
	IPSC	CPC	IPSC	CPC
RAS Capability	Optional	Optional	No	No
Network Application Interface Voice CFS enabled	Optional	Optional	No	No
Wireline Service Enabled (slot 1)	Optional	Optional	No	No
Wireline Service Enabled (slot 2)	Optional	Optional	No	No

Table 9: Service Bit Usage in the Peers in SS/IPSC/CPC system

Peer Service Name	MOTOTRBO Repeater Peer	3rd party Voice/CSBK console
	LCP	LCP
Repeater Call Monitoring	No	Optional
Packet Authentication	Optional	Optional
	Optional	Optional
Slot 1 Phone Gateway	Optional	No
Slot 2 Phone Gateway	Optional	No
RAS Capability	Optional	No
Network Application Interface Voice CFS enabled	Optional	No
Wireline Service Enabled (slot 1)	Optional	No
Wireline Service Enabled (slot 2)	Optional	No

Table 10: Service Bit Usage in the Peers in LCP system

Peer Mode Name	MOTOTRBO Repeater Peer	RDAC Peer	3rd party Voice/CSBK console
	LCP	LCP	LCP
Slot 2 Assignment[1]	Optional	No	No
Slot 2 Assignment[2]	Optional	No	No
Slot 1 Assignment[1]	Optional	No	No
Slot 1 Assignment[2]	Optional	No	No
Peer Status [1]	Optional	No	No
Peer Status [2]	Optional	No	No
Master Peer	Optional	No	No
Remote 3rd Party Application	No	No	Yes
Site Peer	Optional	No	No
No Link Establishment with Data Revert Repeater	No	No	Yes
MOTOTRBO Network Interface Service	No	No	No

Table 11: PeerMode Bit Usage in the Peers in LCP system

Note: For XNL related bit definition, refer to [3]. Here the “optional “means “Depend on the XNL connection status”. For call monitoring bit definition, refer to [5].

In Table 9-Table 11, “No” means the bit must set to 0, “Yes” means the bit must set to 1, “Optional” means the bit can be set to either No or Yes. For example: the packet authentication bit must set to “Yes” if the end user configure the authentication string in the MOTOTRBO CPS; otherwise it must set to “No”. For the packet authentication bit, all the peers in one system must have the same setting. Another example is the repeater call monitoring bit must set to “Yes” if the third party application would like to get the logging messages from the repeater peers.

For the Master Peer bit, only one of the peers in the system can set this bit to Yes. In LCP or CPC system, when a repeater peer finds out there is no Site Peer in the Master’s system map, and it initiates the LE registration on behalf of the Site Peer.

The third party application shall keep the LE connection with the MNIS peer, but shall not initiate a call to it. The voice console application can discover the MNIS peer through the bit setting in the peerMode or peerService field in the LE messages from the MNIS peer.

In Table 9 to Table 11, it is assumed that the RDAC peer provides only RDAC related functions, and the third party application in provides only voice/CSBK related functions. It is possible to have a third party application to have both RDAC and voice/CSBK related functions. In this case, the following rules shall be used:

- No + No = No

- No + Optional = Optional
- No + Yes = Yes
- Yes + Optional = Yes
- Yes + Yes = Yes

For example, the XNL Slave Service bit must set to Yes, and the Repeater Call Monitoring bit can be Optional for the application with both RDAC and voice/data/CSBK functions.

3.1.2 Staggering LE Keep Alive Message in LCP System

After Link Establishment connection, the third party application sends/receives LE Keep Alive messages with all the peers in the system, including RDAC, MNIS and all the repeater peers. A 15-site LCP system can have up to 120 trunking repeaters. If a large number of repeaters send Keep Alive message to the third party application within short period of time, this could interfere with the real time voice traffic processed by the third party application. This can be prevented if the third party application staggers the sending of the Keep Alive Request messages to the repeater peers, and the receiving of the Keep Alive Response from the repeater peers is therefore paced. The repeater peer does not send the Keep Alive Request message the third party application if it sends the Keep Alive Response before its internal Keep Alive Request message timer expires. See Figure 19 for the example message sequence.

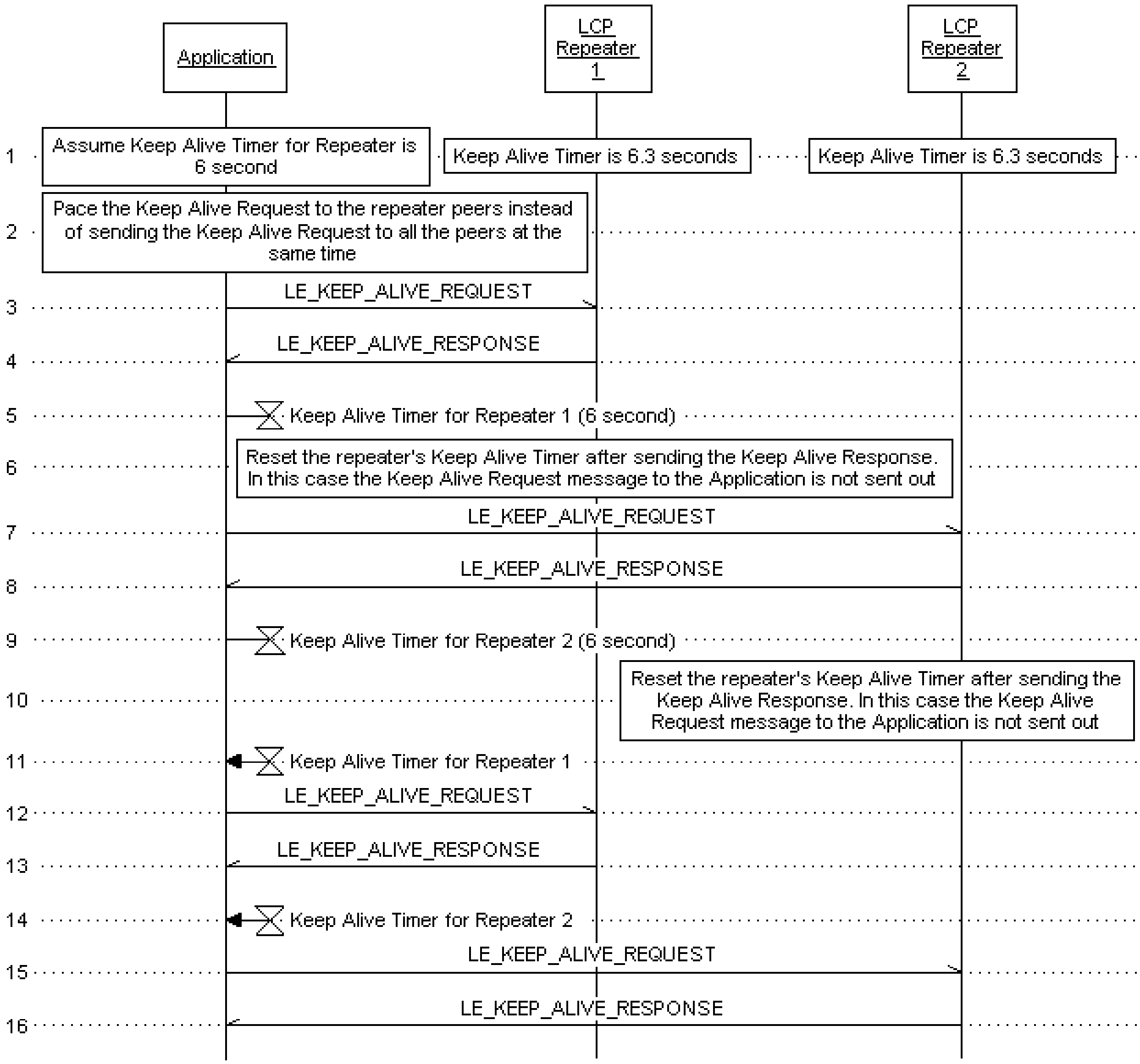


Figure 19: LE Keep Alive Message Staggering

3.2 *Wireline Registration*

A radio only accepts a call if the target individual ID matches it's Radio ID or the target group ID is in the receiving group list. Similar to the radio, the Wireline Registration is intended for the third party application to register the calls which it is interested to receive from the repeater peers as well as identify the application peer of its Radio ID(s) and type of call supported. Without the Wireline Registration, the third party application cannot receive and send Voice/CSBK call.

The Call Control interface is a chargeable feature at repeater. In MOTOTRBO CPS there are two purchasable features at the Network Application Interface:

- Network Application Interface (NAI) Voice
- Network Application Interface (NAI) Data

The table below shows the call types enabled by each feature after purchase.

Network Application Interface Voice	Network Application Interface Data
Bi-direction Voice Call	**Bi-direction Data Call through MNIS interface**
Bi-direction CSBK Call	**Bi-direction CSBK Call**

Table 12: Call Type Enabled by the Network Application Feature

Table 13 shows which NAI feature is required to support the application type. “X” means required; “O” means either NAI Voice or NAI Data, but not both. For example if the third party application only intends to use the bi-direction CSBK call, the customer only needs to either purchase the NAI Voice or NAI Data. If in the future the customer plans to also use the third party application with audio recording, the NAI Voice feature shall be purchased at the first place.

Application Type	Network Application Interface Voice	Network Application Interface Data
Data Call Only		**X**
CSBK Call Only	**O**	**O**
Voice Call Only	**X**	
Audio recording Only	**X**	
Data + CSBK Call		**X**
Data + Voice Call	**X**	**X**
Data + Audio recording	**X**	**X**
Voice Call + CSBK Call	**X**	
Audio recording + CSBK Call	**X**	

Table 13: Required NAI Feature Purchase per Application Type

The third party application needs to check the “Network Application Interface Voice CFS enabled” in the Peer Services Bit Field of the messages from the LE registered repeater peer to determine if the voice and CSBK call interface has been enabled in the repeater before starting the Wireline Registration step. The repeater peer ignores the Wireline Registration message if the Call Control interface has not been enabled. The third party application must conduct the Wireline Registration procedure with all the repeaters which has the “Network Application Interface Voice CFS enabled” in the system. The third party application must still maintain the LE connection with the repeaters that the third party application does not perform Wireline Registration, e.g. data revert repeaters.

Once the third party application finishes the Wireline Registration procedure with the repeater peer, the “Wireline Service Enable (slot 1)” and “Wireline Service Enable (slot 2)” bit in the Services bits are set by the repeater peer. See Table 9 and Table 10 for the detailed definition. In case the LE connection is lost between the repeater peer and the third party application, the application needs to perform the Wireline Registration procedure again with the repeater peer after the LE is reconnected.

The third party application may select monitor service when it wants to receive a call belonging to a radio other than itself. For example a voice recorder may select both monitor bit and voice bit setting with its registration. If it is also intended to monitor CSBK for interested radio ID, it also sets CSBK bit. Monitor is ONLY applicable when an INDIVIDUAL call is being subscribed.

Note: In R2.2 version, the Wireline Registration will not support more than 16 registration entries for radio addresses and 16 registration entries for talk group addresses.

The key components in a Wireline Registration entry are:

- Address Type (Group, All Wide Area Group, All Local Area Group, Individual, All Individual)
- Start Address and End Address (individual ID when the Start Address equals to the End Address, or a range of IDs)
- Call Type (Voice, CSBK)
- Call Attribute (Monitoring, only applicable for individual call)

The following are some examples that can be supported by the Wireline Registration:

- One Console Application with One Radio ID: The console application registers its own individual Radio ID with all the repeater peers. It can also register the group call list. The repeater peers only route the calls which matches the console application’s Radio ID or fits in the console application group list. When the console application initiates a voice or CSBK call to a radio, the console application’s Radio ID will be shown in the receiving radio. The following are the example Wireline Registration entries when the console has radio ID 1 and are interested in group call 1 - 10:

  Entry 1:

    Address Type = Individual

Start Address = 1

End Address = 1

Voice Attributes = %10000000 (bit 7 = registered voice service, bit 6 = normal service)

CSBK Attributes = %10000000 (bit 7 = registered CSBK service, bit 6 = normal service)

Entry 2:

Address Type = Group

Start Address = 1

End Address = 10

Voice Attributes = %10000000 (bit 7 = registered voice service, bit 6 = normal service)

CSBK Attributes = %10000000 (bit 7 = registered CSBK service, bit 6 = normal service)

- Multiple Console Applications in One System: Each console application peer has a different Radio ID. Through the Wireline Registration, the repeater only routes the calls to a specific console application with the matched Radio ID.
- One Console Application with Multiple Radio IDs: It is possible that one console application to have multiple Radio IDs which stand for different support agents or console desks. The Console Application can register multiple Radio IDs or a range of Radio ID. The repeater peer routes the calls to the Console Application as long as the ID matches the registration. The following is the example Wireline Registration entry when the Console has Radio ID 1 – 5.

Entry 1:

Address Type = Individual

Start Address = 1

End Address = 5

Voice Attributes = %10000000 (bit 7 = registered voice service, bit 6 = normal service)

CSBK Attributes = %10000000 (bit 7 = registered CSBK service, bit 6 = normal service)

- Recording Application: If the Application wants to record all the audio, and CSBK call activities, it can register the following two entries:
    - One entry with the address type as All Radio Ids, the voice attributes as voice registration and voice monitoring, and the CSBK attributes as CSBK registration and CSBK monitoring.
    - The other entry with the address type as All Talkgroups, the voice attributes as voice registration.

    The following are the example Wireline Registration entries to record all the audio and CSBK activities.

    Entry 1:

    Address Type = All Individual

    Start Address = 0

    End Address = 0

    Voice Attributes = %11000000 (bit 7 = registered voice service, bit 6 = voice monitor service)

    CSBK Attributes = %11000000 (bit 7 = registered CSBK service, bit 6 = CSBK monitor service)

    Entry 2:

    Address Type = All Group

    Start Address = 0

    End Address = 0

    Voice Attributes = %10000000 (bit 7 = registered voice service, bit 6 = normal service)

    CSBK Attributes = %10000000 (bit 7 = registered CSBK service, bit 6 = normal service)

If the Application wants to only record all individual call in the LCP system, it only needs to register one entry to specify the address type as All Individual and voice attributes as voice registration and voice monitoring.

In the radio network, it is required that each radio has a unique Radio ID. The third party application has to make sure its Radio ID is unique with each of the radio and the other application peers.

After the LE registration procedure finishes, the Wireline Registration procedure needs to start before an application can send/receive voice/CSBK calls using the Call Control messages. The third party application uses the WL_REGISTRATION_REQUEST to subscribe for calls that are of its interest. In the registration request there can be multiple subscriptions for the call IDs and related call services. MOTOTRBO repeaters use the WL_REGISTRATION_STATUS to respond with the result. Using the WL_REGISTRATION_GENERAL_OPS message, the third party application can deregister the calls which it has subscribed to. The result can be gotten either from the WL_REGISTRATION_STATUS or the LE_KEEP_ALIVE.

In the Single Site slots and IPSC local area slot, the Wireline Registration can be set independently in each slot. In the IPSC wide area slot, the Wireline Registration setting has to be the same for all the repeater peers in the same wide area slot. But the setting can be different at two wide area slots.

In CPC and LCP system, the Wireline Registration setting is per repeater peer instead of per slot. In CPC and LCP system, the Wireline Registration setting shall be the same for all the trunking repeater peers at the same site. In LCP system the Wireline Registration setting can be different at different sites. The registrationSlotNumber for CPC and LCP system is recommended to use 0x03.

3.2.1 Call Routing based on Wireline Registration.

When a repeater receives a call over the air or from the phone patch interface, it only sends the call to the third party application peer with matched registration profile:

- For individual call: the target ID fits in the individual radio ID range registered by the third party application. The call attribute type can be monitor or non-monitor.
- For group call: the target group ID fits in the receiving group ID range registered by the third party application.

When a third party application initiates a voice call the repeater peer checks if the source ID in the call request has been registered with the voice call service. If it has not, the repeater peer rejects the call request.

Table 14 summarizes the call routing policies among the radio, the third party application and digital phone patch which connects at the repeater interface. It is possible that there are more than one third party applications joining the same repeater system. The repeater routes the call to all the third party applications as long as they subscribe for the call.

Call Originator	Call Type	Call Receiver		
		Radio	Third Party Application(s)	Digital Phone Patch
Radio	**Individual Voice / CSBK**	Only if the target ID matches the receiving Radio's ID	• If the target ID matches the individual radio ID registered by the third party application OR • Wildcard address is registered by the third party application	Only if the call is a phone call with the correct access code
	Group Voice	Only if the group ID is in the receiving group list	• If the group ID matches the group ID registered by the third party application OR • Wildcard address is registered by the third party application	Only if the call is a phone call with the correct access code
Third Party Application	**Individual Voice / CSBK**	Only if the target ID matches the receiving Radio's ID	• If the target ID matches the individual radio ID registered by the third party application OR • Wildcard address is registered by the third party application	NA
	Group Voice	Only if the group ID is in the receiving group list	• If the group ID matches the group ID registered by the third party application OR • Wildcard address is registered by the third party application	NA
Digital Phone Patch	**Individual Phone Call**	Only if the target ID matches the receiving Radio's ID	• If the target ID matches the individual radio ID registered by the third party application OR • Wildcard address is registered by the third party	NA

Call Originator	Call Type	Call Receiver		
		Radio	Third Party Application(s)	Digital Phone Patch
			application	
	Group Phone Call	Only if the group ID is in the receiving group list	• If the group ID matches the group ID registered by the third party application OR • Wildcard address is registered by the third party application	NA

Table 14: Call Routing Policies

3.2.2 Initial Wireline Registration

This section gives two examples on the initial Wireline Registration. Section 3.2.2 shows update on the existing Wireline Registration and section 3.2.4 shows Wireline De-registration.

Use Case 1: A third party application works with 2 radios in a MOTOTRBO repeater system. It subscribes 3 entries and the last entry is only for voice monitoring purpose (both voice bit and monitoring bit are set). The application initiates calls whose source radio IDs are in its 1st entry in registration request profile. The application joins group calls which are in its 2nd entry. The application receives calls which are targeted at the radio ID in both its 1st entry and 3rd entry in the registration request profile.

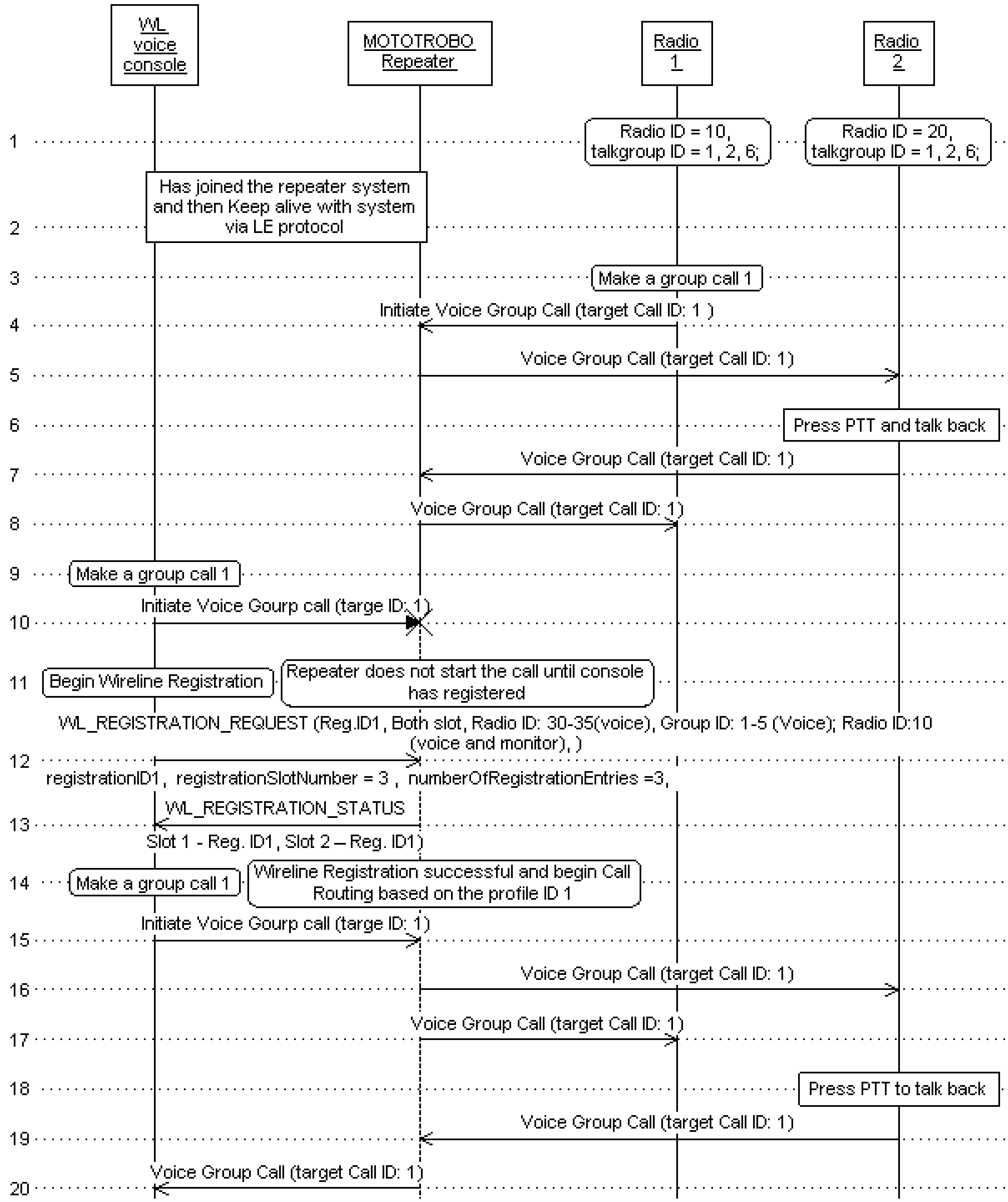


Figure 20: Call Routing based on Wireline Registration

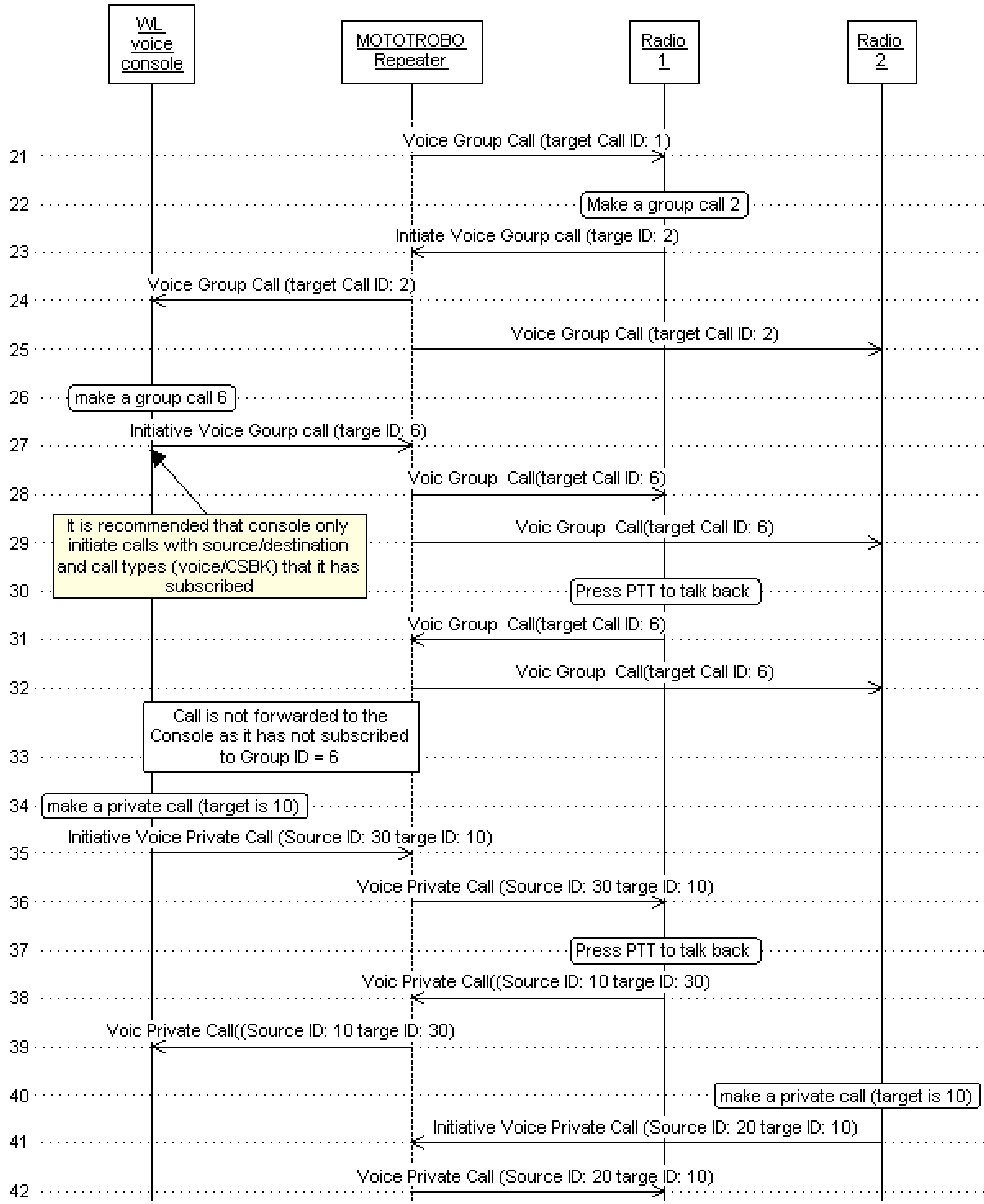


Figure 21: Call Routing based on Wireline Registration (Cont)

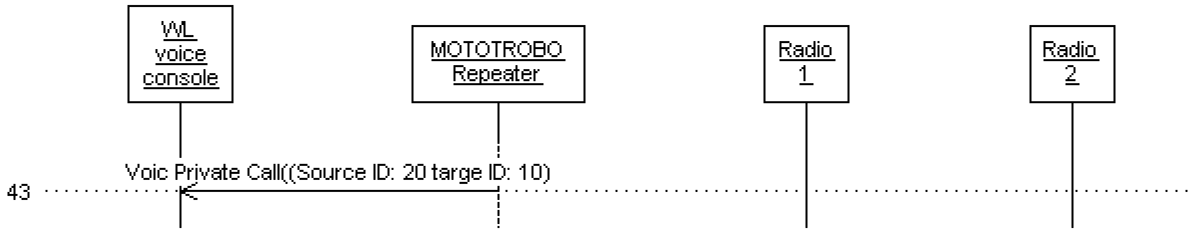


Figure 22: Call Routing based on Wireline Registration (Cont)

Use Case 2: A third party application works with 2 radios in a MOTOTRBO repeater system. It subscribes 2 entries both as wildcard.

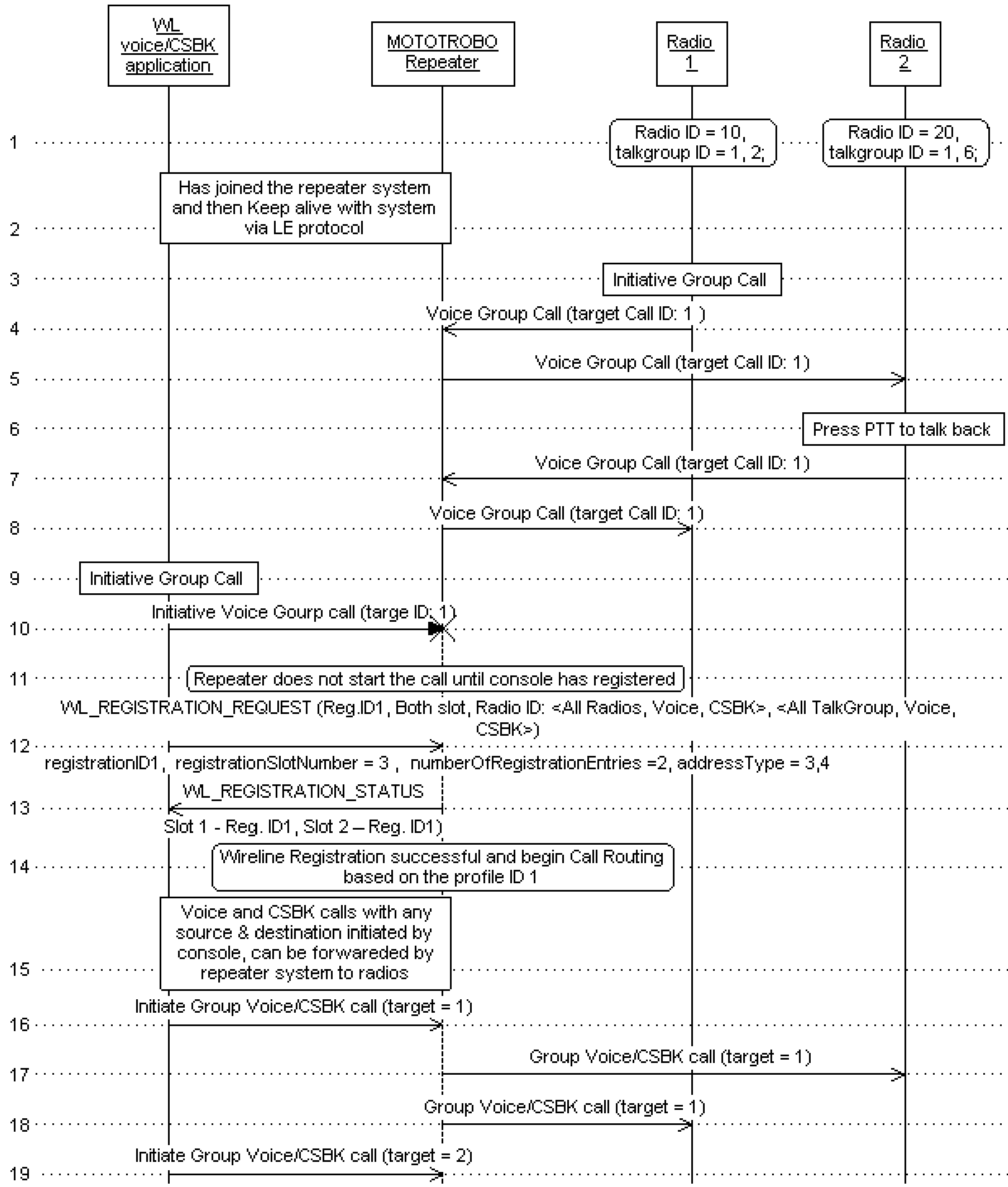


Figure 23: Wildcard Call Routing based on Wireline Registration

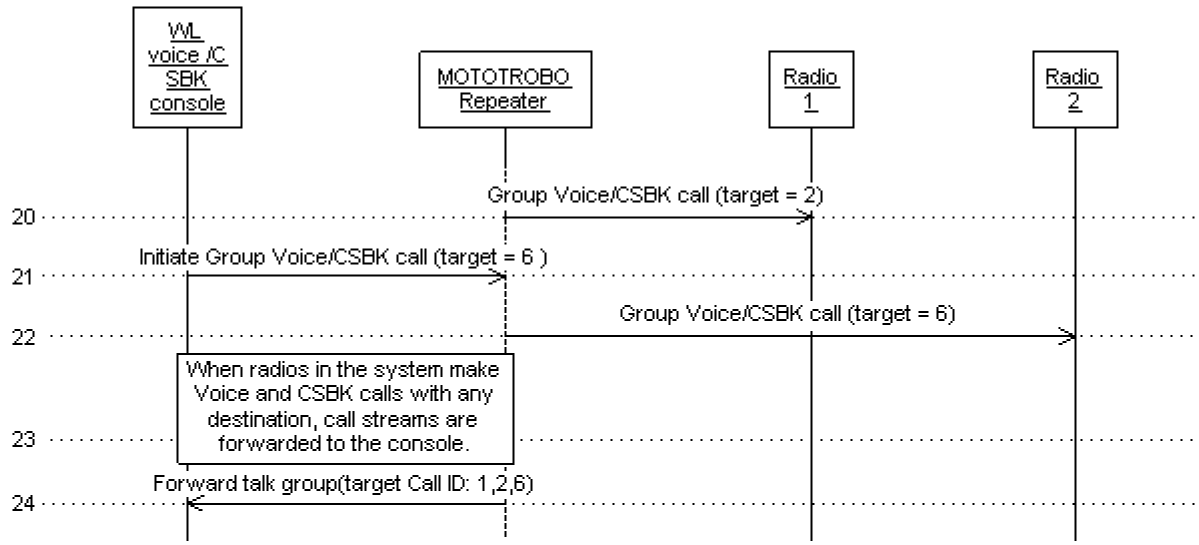


Figure 24: Wildcard Call Routing based on Wireline Registration (Cont)

3.2.3 Wireline Registration Change

The third party application can query the existing registration profile using WL_REGISTRATION_GENERAL_OPS. It can change the registration profile based on the registration ID in the WL_REGISTRATION_REQUEST message.

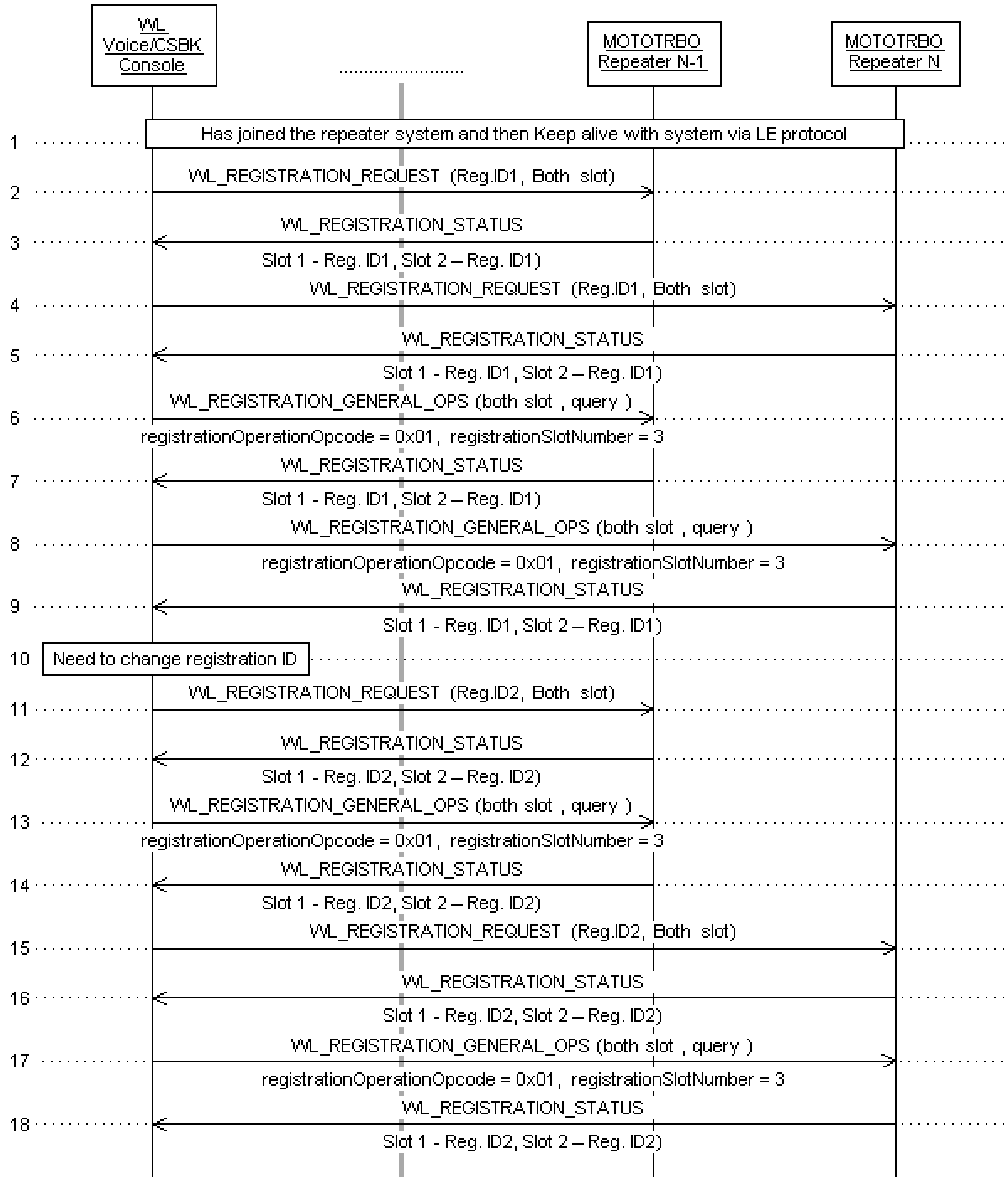


Figure 25: Wireline Registration Change

3.2.4 Wireline Deregistration

The third party application can de-register the certain registration profile based on the registration ID through the WL_REGISTRATION_GENERAL_OPS message. Upon receipt of this message, the repeater peer responds with a Wireline Registration Status. After that the LE KeepAlive message from the repeater peers also indicates the Wireline Service as Disabled for that slot or repeater.

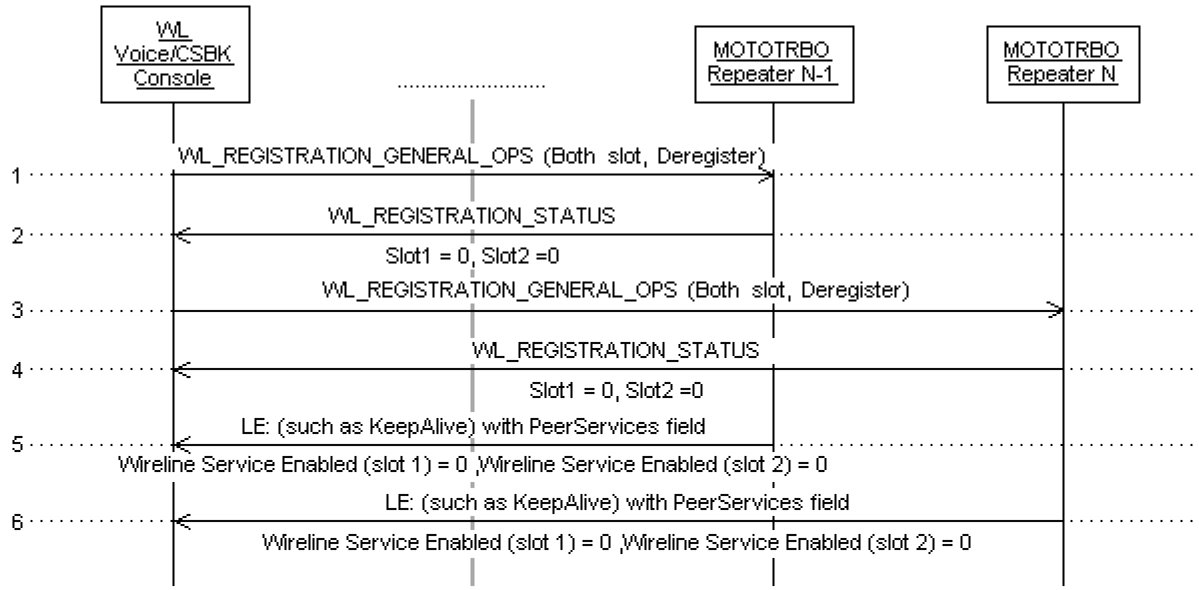


Figure 26: Wireline Deregistration

3.2.5 Wireline Registration in Single Site / IPSC System

In Single Site System and IP Site Systems, the third party application may make different Wireline Registration on each slot of the MOTOTRBO repeater. The LE KeepAlive messages from the repeater peer reports the Wireline Service status per slot.

At the wide area slot in an IP Site Connect system, the third party application shall conduct the Wireline Registration with all the repeaters at the wide area slot with the same registration profile.

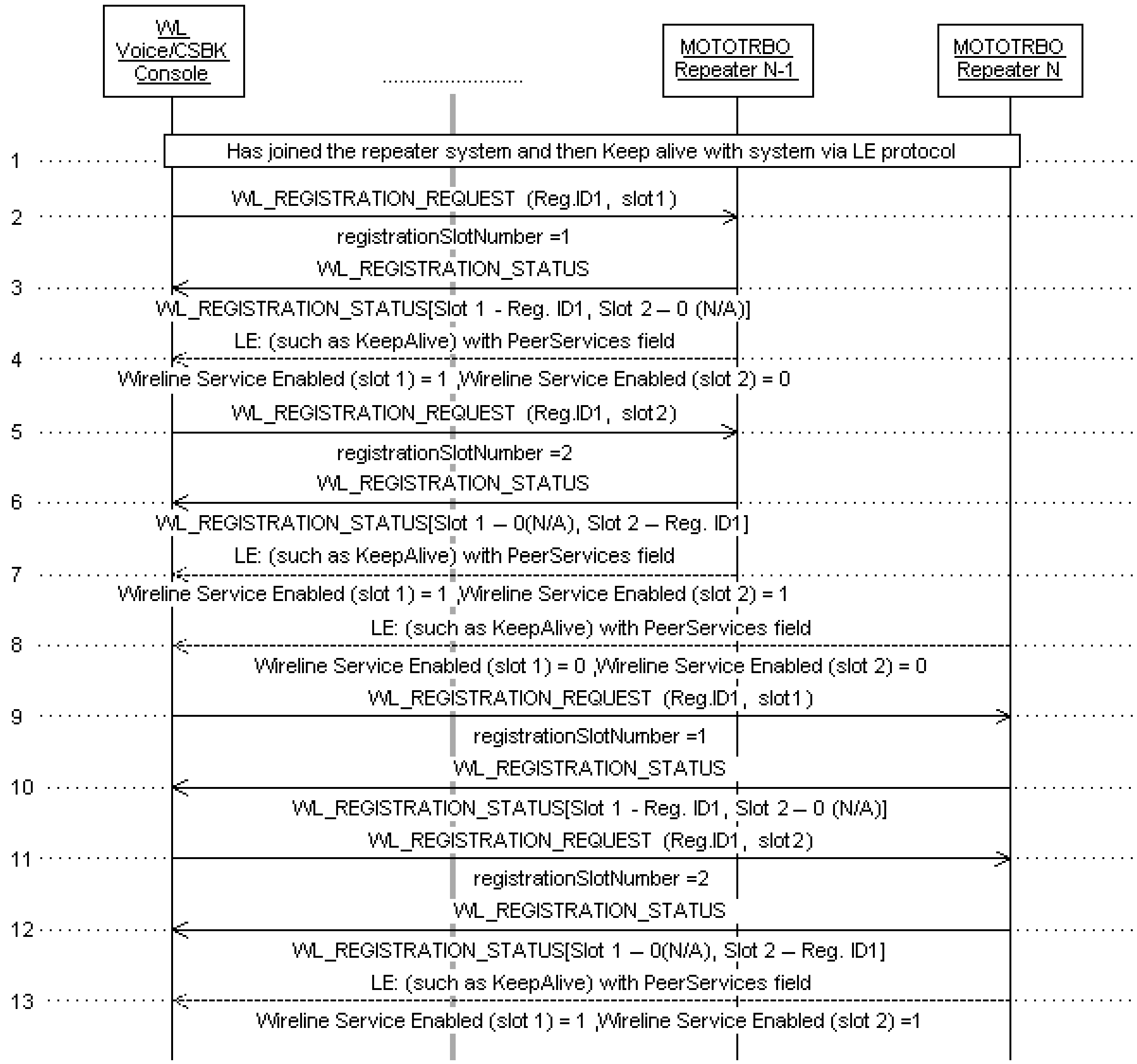


Figure 27: Wireline Registration in Single Site / IPSC System

3.2.6 Wireline Registration in Capacity Plus System

In a Capacity Plus System, the third party application shall conduct the Wireline Registration with all the trunking repeaters with the same registration profile at both slots. The third party application does not need to conduct the Wireline Registration with all the data revert repeaters.

Note: Although the third party application must make LE Registration with the Site Peer (peer ID is 0x00), it does not conduct Wireline Registration with the Site Peer.

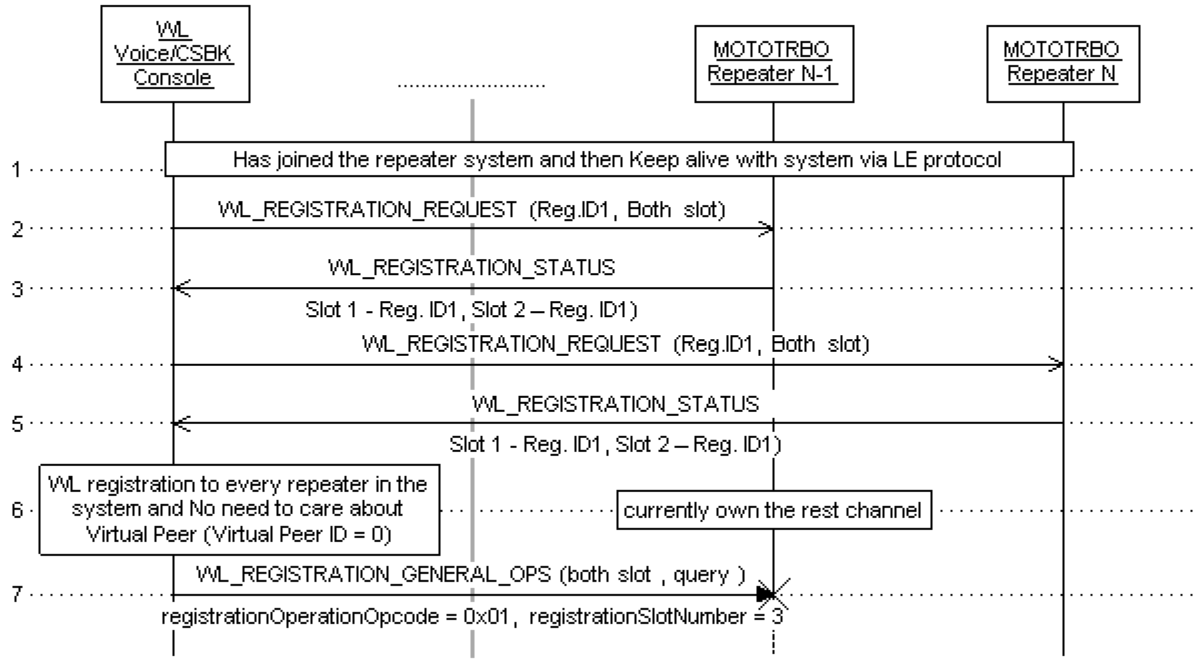


Figure 28: Wireline Registration in Capacity Plus System

3.2.7 Wireline Registration in Linked Capacity Plus System

Similar to Capacity Plus systems, in Linked Capacity Plus systems, the third party application shall have Wireline Registration with all the trunking repeaters at the same site with the same registration profile at both slots, and not with all the data revert repeaters. It can have different registration profile at different sites.

Note: Although the third party application must make LE Registration with the Site Peers (peer ID is 0x00), it does not conduct Wireline Registration with the Site Peers.

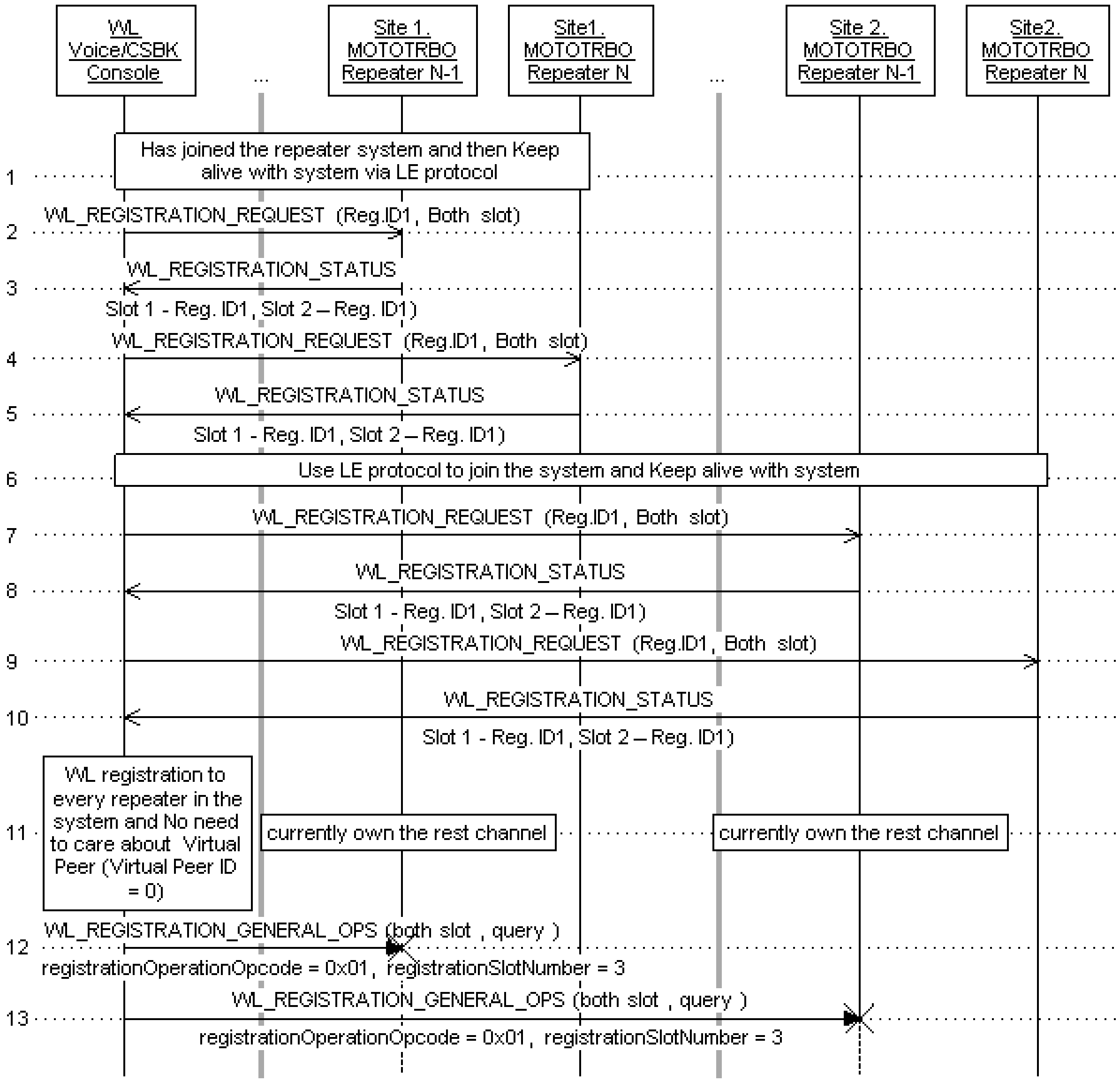


Figure 29: Wireline Registration in Linked Capacity Plus System

3.3 *Wireline Protocol Version*

The wireline versioning is to support backwards compatibility for the applications utilizing the Call Control Protocol. It is to identify the change specific for the Call Control interface. There is a 2-byte Wireline Protocol Version field at the end of all Call Control messages.

The peers use the Wireline Protocol Version fields to communicate their supported protocol range and agree on a common version during Wireline Registration. When the peers have the same current Wireline Protocol Version, they choose the current Wireline Protocol Version. When the peers do not have the same current Wireline

Protocol Version, they compare the versions. If the peer with the greater current Wireline Protocol Version supports the smaller current Wireline Protocol Version used by the other peer, both peers use the smaller current Wireline Protocol Version for the Call Control messaging between these two peers. If the peer cannot support the version range used by the other peer, it rejects the Wireline Registration message from the other peer.

3.3.1 Wireline Protocol Version Definition

The Call Control messages structure is shown as below:

NAI Header	Call Control PDU	Current / Accepted Wireline Protocol Version	Oldest Wireline Protocol Version	Wireline Authentication ID	Wireline Authentication Signature

Figure 30: Wireline Protocol Version field location in the PDU

The Wireline Protocol Version field definition is as below:

WL Protocol Version Bits	WL Protocol Version Name	Description
0-1	Minor Version Information	%00$_2$ - reserved for future expansion
2-7	Major Version Information	%000001$_2$ = Major Version %000010$_2$ - %100000$_2$ - reserved for future expansion

Table 15 – Wireline Protocol Version Field Definition

3.3.2 Wireline Protocol Version Negotiation

Although the Wireline Protocol Version Negotiation rule is similar as the LE Version Negotiation, they are totally different and have no relationship. Wireline Protocol Version fields are only used in the Call Control messages while the LE protocol version fields are only used in the LE messages. The Wireline Protocol Version Negotiation shall begin after the LE registration procedure completed and the third party application has joined the repeater system already.

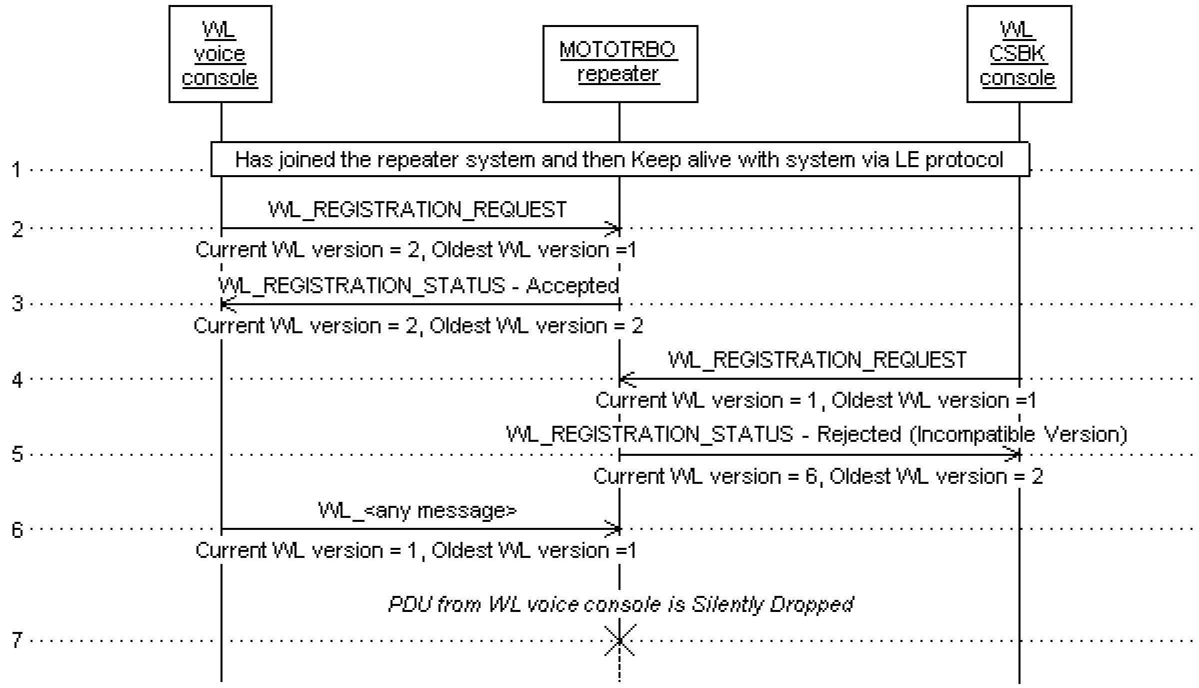


Figure 31: Wireline Protocol Version Negotiation

3.4 *Wireline Authentication*

At the NAI interface, there is an optional configuration schema to support the HMAC-SHA-1 protocol authentication. See the section of Optional NAI LE Authentication Footer in the Reference [1] for more details. To further protect the access to the NAI Call Control interface, the Wireline Authentication is introduced. The Wireline Authentication is a one-way authentication which only applies to those Call Control messages in Table 1 from the third party application peer(s) to the Repeater peer(s). The message sent from the Repeater peers(s) to the third party application peer(s) does not have the Wireline Authentication signature.

3.4.1 Wireline Authentication Footer

As shown in Figure 32, the NAI Authentication is optional. If the NAI authentication key is not configured in the peers, the 80-bit NAI authentication key does not present at the end of all NAI messages. However, the Wireline Authentication Footer block is mandatorily required to be present in all the Call Control PDUs from the third party application peer to Repeater peer(s) regardless the NAI authentication key is configured or not.

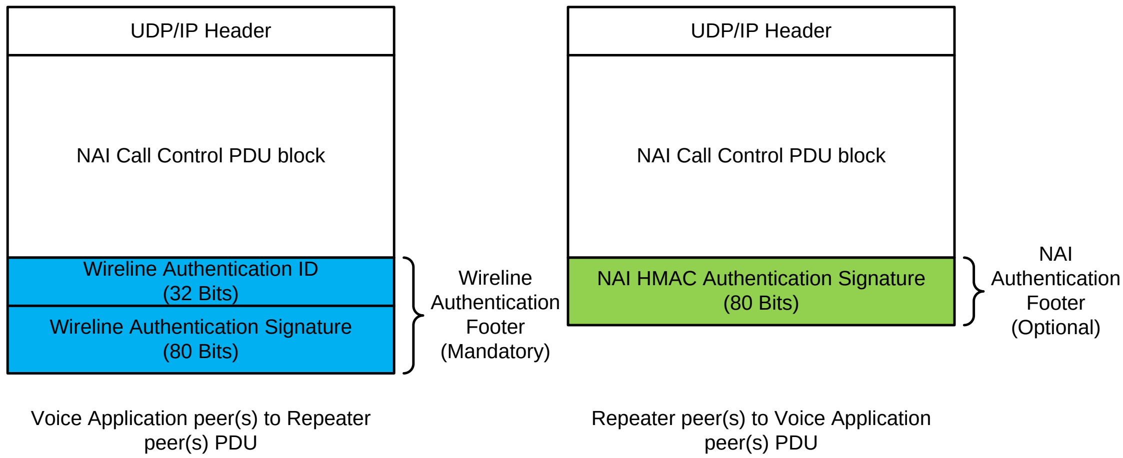


Figure 32: NAI Authentication and Wireline Authentication Footer Structure

Each 3rd party developer is assigned uniquely with a pair of 32-bit Wireline Authentication ID and 160-bit Wireline Vendor Key. The third party developers have to contact the regional manager to get the Wireline Authentication ID and Vendor Key. The Wireline Authentication and Vendor ID are Motorola confidential information, and the third party application shall protect the key information as much as possible.

The Wireline Authentication ID is required to be filled in the Wireline Authentication Footer block, along with the Wireline Authentication Signature of the Call Control PDU. The section below provides detailed on how to compute the Wireline Authentication Signature.

3.4.2 Wireline Authentication Key and Signature

Similar to the NAI HMAC Authentication Signature, the Wireline Authentication Signature is also computed via SHA-1/HMAC algorithm and truncated to 80 bits. The secret key used to compute the signature, i.e. Wireline Authentication Key (Kw), is 320 bits long. The Wireline Authentication Key is constructed as below:

kv = 160 bits Wireline Vendor Key;
kc = 160 bits right-aligned P2P HMAC Authentication Key with zero padding in the left.
Kw = [kv kc] = [kv1 kv2 kv3…kv160 kc1 kc2 kc3…kc160]

Figure 33 shows the Kw generation in the cases of NAI authentication key is configured and NAI authentication key is not configured.

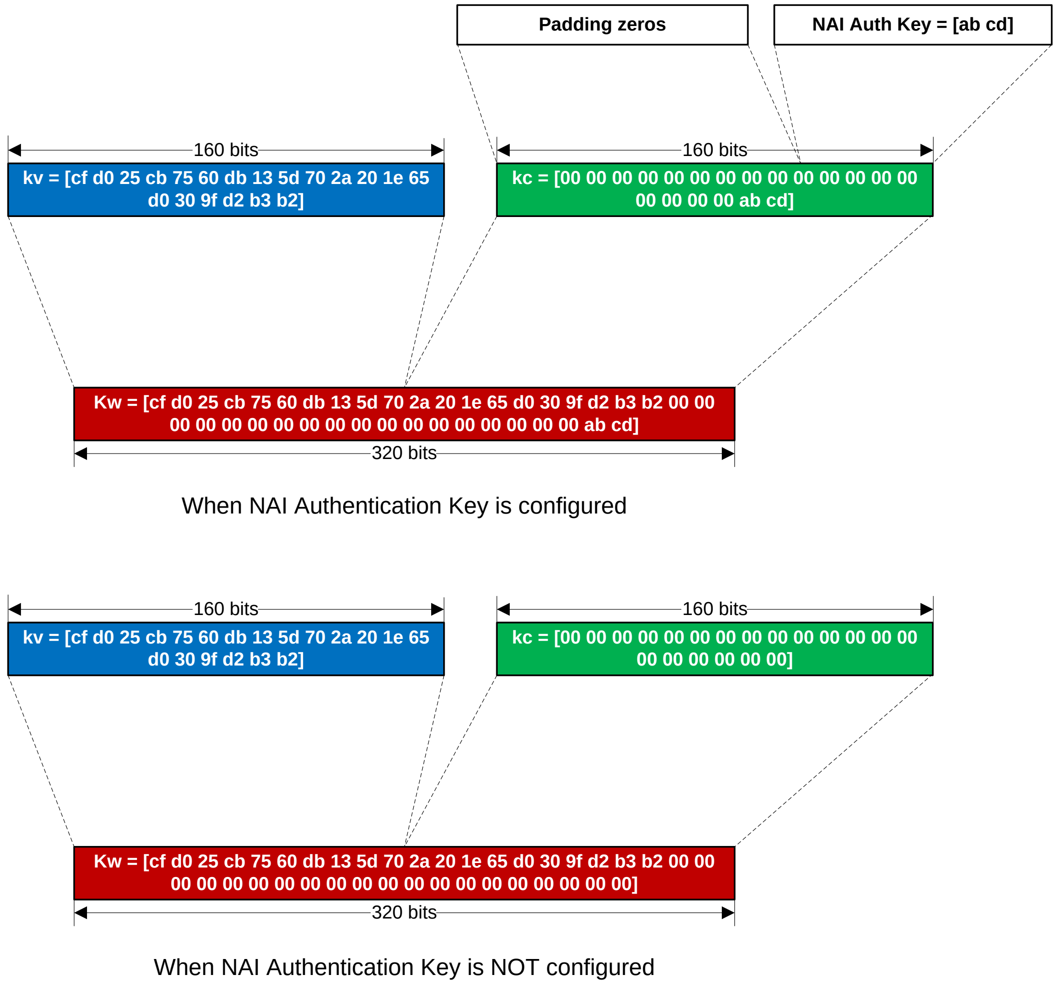


Figure 33: Wireline Authentication Key Structure

The Kw is used as the secret key to hash the Call Control PDUs with the SHA-1/HMAC algorithm. Note that only the Call Control PDU block shall be hashed. The Call Control PDU block starts with the NAI Opcode 0xB2. Neither the IP/UDP header nor the Wireline Authentication Footer block shall be hashed. The 160-bit output is truncated to 80 bits (MSBs) and filled to the Wireline Authentication Signature field.

Below example can be used by the 3rd developers to verify their Wireline Authentication implementation.

Wireline PDU Block = b2 00 00 00 01 01 01 94 b8 cf 00 00 01 00 01 01 00 00 00 01 00 00 00 01 00 00 80 04 04

Wireline Authentication ID = 01 01 00 2a
Wireline Vendor Key (kv) = cf d0 25 cb 75 60 db 13 5d 70 2a 20 1e 65 d0 30 9f d2 b3 b2

ALT 1: NAI HMAC Authentication key (Kc) is configured as 0xab 0xcd;

Kc = 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ab cd

Wireline Authentication Key (Kw) = cf d0 25 cb 75 60 db 13 5d 70 2a 20 1e 65 d0 30 9f d2 b3 b2 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ab cd

Wireline Authentication Signature = **c0 c5 d7 59 4b 6b 85 61 a2 8e**

Complete Wireline PDU = b2 00 00 00 01 01 01 94 b8 cf 00 00 01 00 01 01 00 00 00 01 00 00 00 01 00 00 80 04 04 01 01 00 2a **c0 c5 d7 59 4b 6b 85 61 a2 8e**

ALT 2: NAI HMAC Authentication is NOT configured;

kc = 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Wireline Authentication Key (Kw) = cf d0 25 cb 75 60 db 13 5d 70 2a 20 1e 65 d0 30 9f d2 b3 b2 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Wireline Authentication Signature = **b7 10 d0 48 10 4c 2a be 13 3b**

Complete Wireline PDU = b2 00 00 00 01 01 01 94 b8 cf 00 00 01 00 01 01 00 00 00 01 00 00 00 01 00 00 80 04 04 01 01 00 2a **b7 10 d0 48 10 4c 2a be 13 3b**

3.5 *Wireline Voice Call*

3.5.1 Call Stream and Call Session

Before we introduce the detailed voice call setup procedure, the following key concepts deserve a clear definition:

- **Call Stream:** One audio flow originated from an end user. It starts with the PTT press and ends with the PTT release from the same end user.
- **Call Session:** A voice call which could contains multiple call streams from the end users.
- **Call Session Hang Status:** When the repeater is in Call Hang Time. During this period the call participant can respond, whose call stream is still considered as part of the current call session.
- **Call Session End Status:** When the repeater Call Hang Time expires, the voice call is considered to be over.

The Call Session in Figure 34 contains two call streams: one is originated from radio1 and the other is originated from radio2.

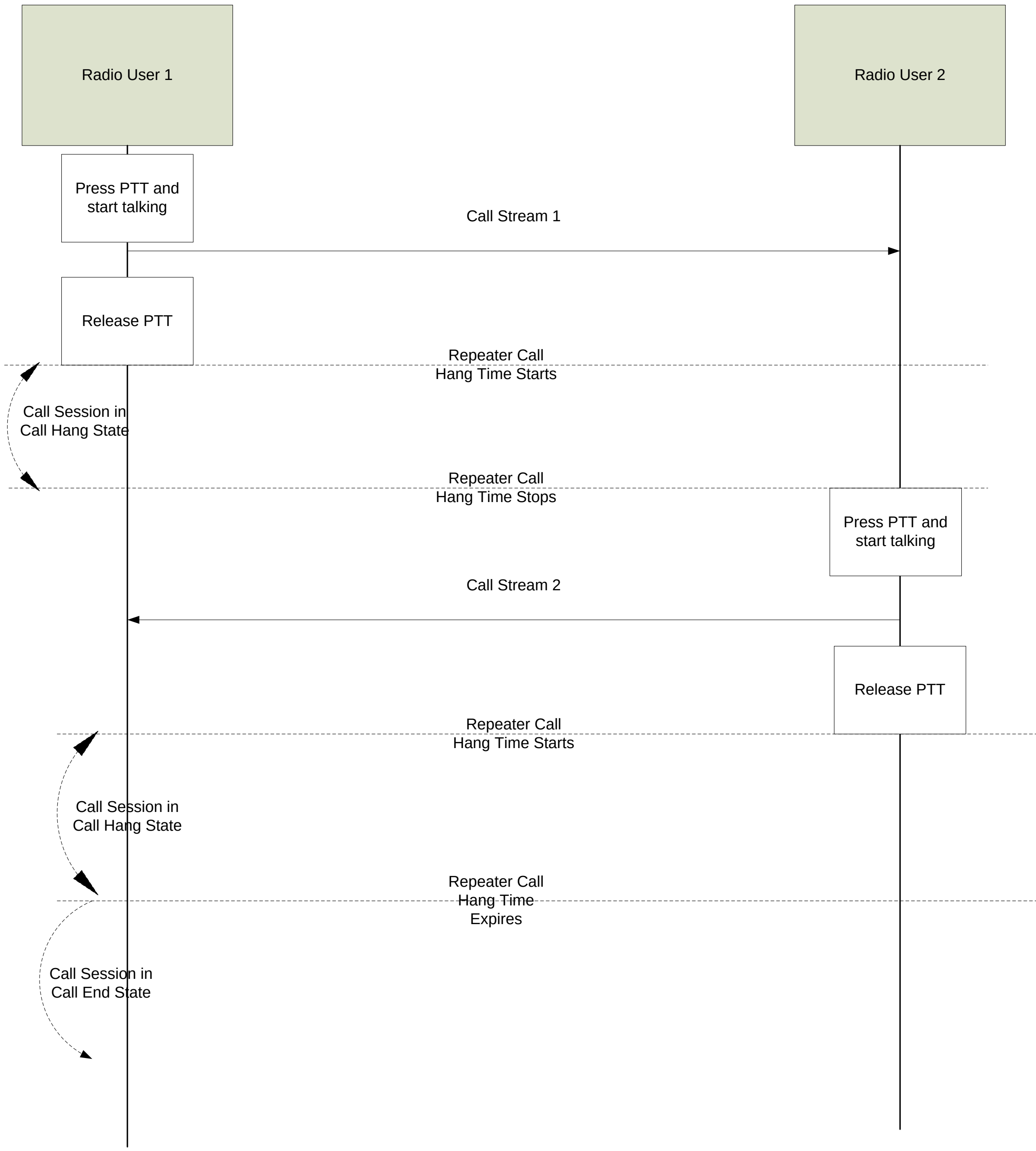


Figure 34: Call Session and Call Stream Definition

3.5.2 Roles in the Voice Call

In general, there are three phases in a voice call:

- Call setup
- Audio streaming

- Call tear down

There are different responsibilities owned by the third party application and repeater peers in those phases. Table 16 summarizes the responsibilities of repeater peer and third party application when radio initiates a call to the third party application or the third party application initiates a call to the radio.

Call Originator	Repeater Peer	Third Party Application
Radio	• Notify the channel status change • Floor arbitration with other repeater peers • Indicate the beginning of a call stream • Stream audio data to the third party and the repeater peer • Generate DMR voice bursts and repeat the call over the air • Indicate the end of a call stream • Maintain the repeater call hang timer • Indicate the call session status	• Retrieve the call information from the Call Control interface message • Extract the audio data from the Call Control interface message • Decode the AMBE audio
Third Party Application	• Notify the channel status change • Update the call status to the third party application, e.g. request received, channel granted • Floor arbitration with other repeater peers • Stream audio data to the other third party application and the	• Check if the channel status is idle before sending the call request to the repeater peer • Re-send the call request if the call is not set up successfully, e.g. lack of response from repeater, other call wins the channel. • Encode audio data into AMBE audio • Stream the AMBE audio data in the Call Control message at 60ms pace

Call Originator	Repeater Peer	Third Party Application
	repeater peer • Generate DMR voice bursts and repeat the call over the air • Indicate the end of a call stream • Maintain the repeater call hang timer • Indicate the call session status	

Table 16: Peer Responsibilities in Voice Call

The third party application shall only initiate a call when the channel status is idle because the repeater peer cannot queue the call request. For trunking systems, the call request is always sent to the Site Peer; the third party application can initiate multiple calls into the system, but make sure initiate the new call request after the first one has been accepted by the repeater system.

3.5.3 Call Arbitration

At the Call Control interface, the call arbitration is transparent to the third party application. The third party application only gets the arbitration result from the repeater peer. However, from the system view, it is beneficent to know how the call is selected by the repeater system.

Based on where the call is received, there are two call categories in the repeater peer:

- Local Call: Received over the air; from the third party application via network interface; or from the phone patch through the 4-wire interface

- Remote Call: Received from a remote repeater peer via network interface.

When more than one “local calls” are received, the repeater peer uses the First-In-First-Out (FIFO) to decide the winner.

When there are both “local calls” and “remote calls” received, the repeater peer uses the FIFO rule to decide the winner of the “local calls” first. The winner of the “local calls” competes with the “remote calls” using the wide area arbitration rules.

The wide area arbitration rule determines the call based on the call priority. The following is the call priority list with descend order in an IP Site Connect system:

1. Emergency Call
2. Voice call from third party application with impolite access criteria
3. Voice Call with polite access criteria
4. Data and CSBK Call

The following is the call priority list with descend order in Linked Capacity Plus system:

1. Emergency Call
2. All Call
3. Voice call from third party application with impolite access criteria
4. Group Call with polite access criteria
5. Individual Call with polite access criteria
6. Data and CSBK Call

3.5.3.1 Call Arbitration During a Call Session

The third party application call request is also arbitrated when the third party application wants to talk back or initiates a call stream during call hang. The third party application call request can also get declined if the radio talks back first. The third party application may take over the radio's call with impolite or interrupt access criteria. See for example message sequence. See section 3.5.10 for more detailed information on impolite and transmit interrupt access criteria.

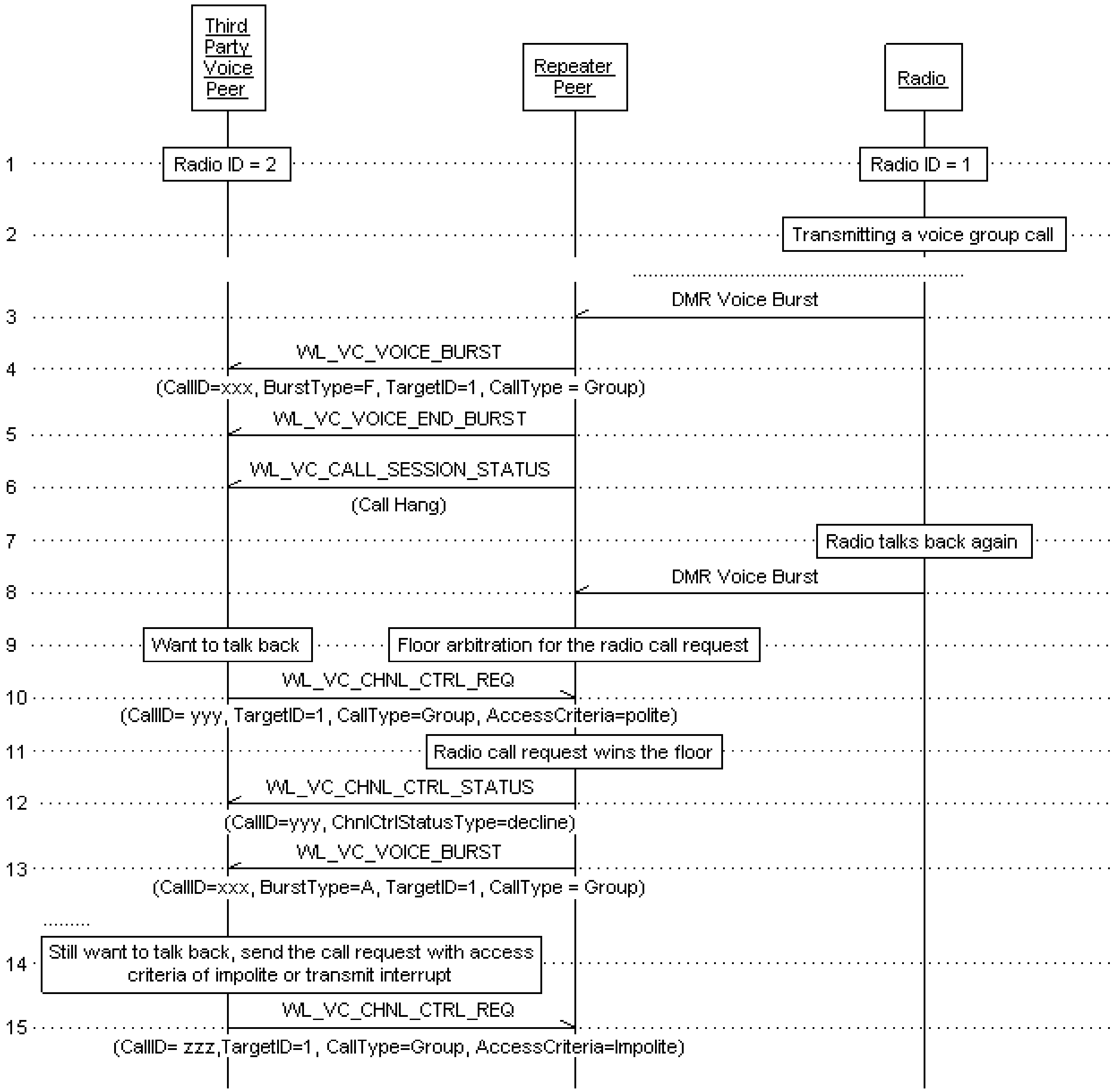


Figure 35: Talk Back During a Call Session

3.5.4 Audio Data Format

Even though the repeater peer is responsible to generate the over the air DMR voice bursts, the third party application has to follow certain pattern to fill in the audio data in the Call Control message. The following sub-section gives a brief overview on the DMR voice burst structure.

3.5.4.1 DMR Voice Bursts

As specified in Reference [5], voice data is transmitted using voice super frames over the air. Each voice super frame contains 6 bursts: A, B, C, D, E, and F shown in Figure 36. Each burst contains three 20ms vocoder compressed frames. Each voice super frame contains 360ms vocoder frames.

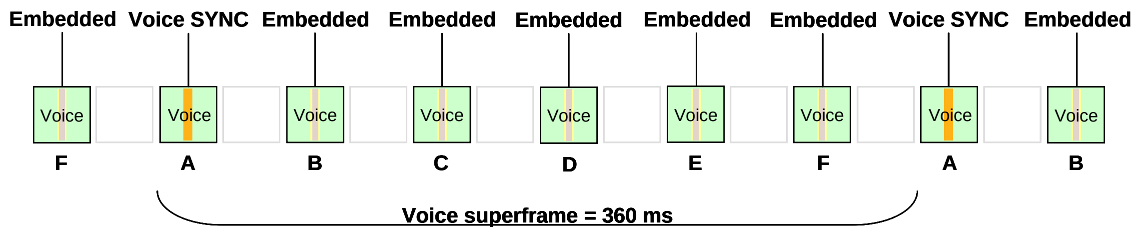


Figure 36: Voice Super Frame

In addition to vocoder bits, voice bursts carry either EMBedded signaling (EMB field + embedded signaling) or frame SYNChronization (SYNC) in the center of the burst. The embedded signaling carries the Link Control (LC) messages such as Group Voice Channel User LC or Unit to Unit Voice Channel User LC. The embedded signaling enables the late join feature.

When a voice call starts, a voice Link Control (LC) header must be transmitted before the first voice super frame as shown in the first burst of Figure 37.

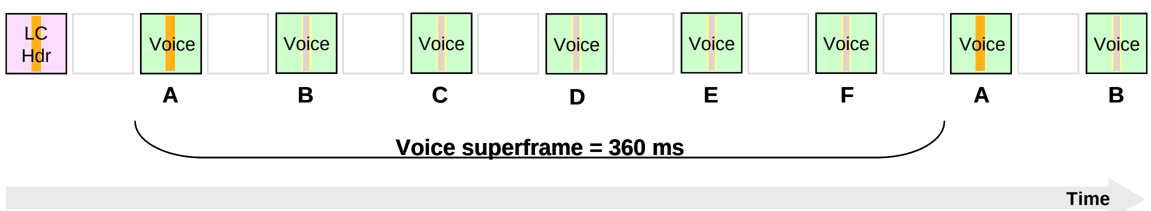


Figure 37: Voice Initiation with LC Header

When a voice call ends, the voice data transmission must be terminated by sending a voice LC terminator after the last voice super frame. The last Data/Control burst in Figure 38 is the voice LC terminator.

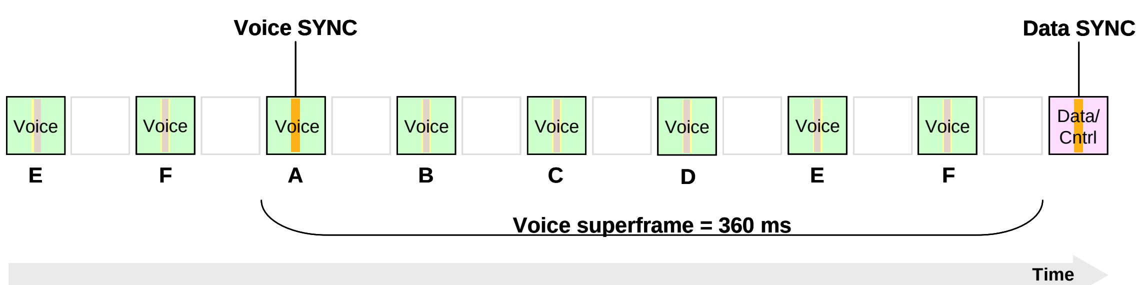


Figure 38: Voice Terminator

The voice LC header and voice terminator contain the same LC message in one audio stream.

3.5.4.2 Call Control Voice Packet

In the Call Control interface, the WL_VC_VOICE_BURST message is used to carry the audio data in both receiving and sending a voice call. Each WL_VC_VOICE_BURST carries 60ms audio data in the field of AMBE voice encoded frames. The field of burstType in the WL_VC_VOICE_BURST message tells if the audio data is for burst A, B, C, D, E or F. The third party application has to maintain the burst sequence when streaming the audio data to the repeater.

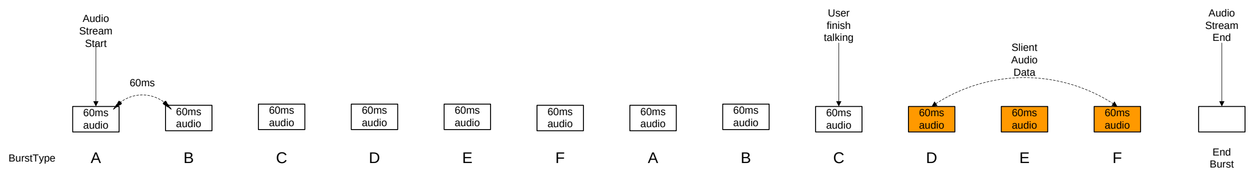


Figure 39: Audio Streaming

The user talking audio may not perfectly align with the DMR voice superframe, the third party application shall finish the voice superframe by filling in the silent audio data. In Figure 39, the WL_VC_VOICE_BURST messages for the D, E, and F in orange burst contain the silent audio data.

The repeater peer is responsible for generating the voice header and voice terminator, and filling the embedded signaling and SYNC frame into the DMR voice bursts when repeating the call over the air.

In summary, when initiating a call, the third party application has to ensure:

- Start to send the WL_VC_VOICE_BURST within 720ms when receiving the WL_VC_CHNL_CTRL_STATUS with status of Granted
- The interval between the WL_VC_VOICE_BURST messages is 60ms.
- Follow the burst type order: A, B, C, D, E and F, and maintain the increment in the RTP header's sequence number field and timestamp field.
- Send the WL_VC_VOICE_END_BURST at the end of the audio stream

3.5.4.3 AMBE + 2 and FEC Encoding / Decoding

The DMR FEC is used to correct errors over the air communication only. FEC is corrected by the MOTOTRBO repeater upon receiving a burst over the air. FEC is applied on the voice data and embedded signaling by the MOTOTRBO repeater before transmitting over the air. Therefore, the third party application does not need to implement the DMR FEC encoding and sending out to all the peers.

The voice frames in the WL_VC_VOICE_BURST message contain 2450bit/second DVSI AMBE+2 encoded voice and 0 bit FEC encoding bit. Each 49-bit AMBE voice frame in the WL_VC_VOICE_BURST message contains 20ms audio. The third party application must apply AMBE+2 encoding/decoding on the voice frame that are sent or received over the IP network.

There are two ways for a third party application to accomplish the proprietary 2450 Hz AMBE+2 vocoder with 0 Hz FEC encoding:

- Obtains a developer's license to AMBE+2 software from DVSI as well as a means of controlling end customer license fees for DVSI.

- Builds a hardware component that has an AMBE-3000 IC or AMBE-3300 purchased from DVSI in the third party solution. For example, create a USB vocoder dongle or embedded network gateway box using the AMBE-3000 IC and connect it to the PC where the third party application runs. Refer to http://www.dvsinc.com/products/a3000.htm or http://www.dvsinc.com/products/a3003.htm for detailed information on the AMBE chips.

3.5.5 RTP Header for the Call Control Voice Packet

Figure 9 shows the protocol stack for the voice call audio streaming part. There is a 12-byte RTP header in the WL_VC_VOICE_BURST and WL_VC_VOICE_END_BURST message. The CSRC field in the RTP header is not used in the Call Control interface. The settings for the RTP header are recommended as:

- Version (Ver): Always Set to $(10)_2$

- Padding (P): Always Set to 0_2

- Extendsion (X): Always Set to 0_2

- CSRC Count (CC): Always Set to 0_2

- Marker (M): Set to 1_2 in the first WL_VC_VOICE_BURST message in a call stream. Set to 0_2 for all the remaining WL_VC_VOICE_BURST message

- Payload Type (PT): Set to 0x5D for all the WL_VC_VOICE_BURST messages from the very first one to the last one in a call stream. Set to 0x5E in WL_VC_VOICE_BURST_END message

- Sequence Number: Can start with any number. It is incremented by one for each WL_VC_VOICE_BURST message.

- TimeStamp: This is a relative timestamp. It is incremented by 480 (60ms * 8000Hz) for each WL_VC_VOICE_BURST message.

- SSRC: Always set to 0

Example of the 12-byte RTP header in the first WL_VC_VOICE_BURST message:

80 dd 0b 65 00 00 00 00 00 00 00 00

Example of the 12-byte RTP header in the second WL_VC_VOICE_BURST message:

80 5d 0b 66 00 00 01 e0 00 00 00 00

Example of the 12-byte RTP header in the WL_VC_VOICE_BURST_END message:

80 5e 6b 00 00 0b 40 00 00 00 00

3.5.6 Example Call Control Voice Packets

This section shows some of the example messages used in the voice call setup and call streaming.

Table 17 shows WL_VC_CHNL_CTRL_REQUEST message for a third party application to set up a clear group call in a trunking system with polite access.

Offset	Raw Data	Description
0	0xb2	Call Control Opcode
1	0x00000264	Application Peer ID
5	0x13	Opcode for WL_VC_CHNL_CTRL_REQUEST
6	0x00	Slot Number. The repeater will assign a slot number for the call
7	0x4c dc 12 38	Call ID for this call
11	0x4F	Call type: group voice call
12	0x00 00 00 01	Source ID
16	0x00 00 01 94	Target ID
20	0x01	accessCriteria is polite
21	0x00	callAttribute is clear call
22	0x00	Reserved field
23	0x00	Preamble Duration. For voice call set it to 0.
24	0x00 00	Reserved
26	0x00 00 00 00 00 00 00 00	CSBK Parameters: Not needed for voice call
34	0x04	Current / Accepted Wireline Protocol Version
21	0x04	Oldest Wireline Protocol Version
22	xx xx xx xx	Wireline Authentication ID
26	xx xx xx xx xx xx xx xx xx xx	Wireline Authentication Signature

Table 17: Example WL_VC_CHNL_CTRL_REQUEST Message

Table 18 shows the very first WL_VC_VOICE_BURST message for a third party application to stream a clear group call with polite access.

Offset	Raw Data	Description
0	0xb2	Call Control Opcode
1	0x00 00 03 e9	Application Peer ID
5	0x21	Opcode for WL_VC_VOICE_BURST
6	0x02	Slot Number: assigned by the repeater peer
7	0x4c dc 12 38	Call ID for this call
11	0x4f	Call Type: group voice call
12	0x00 00 00 01	Source ID
16	0x00 00 01 94	Target ID
20	0x00	callAttributes: clear call
21	0x00	Reserved
22	*0x80 dd 0b 65 00 00 00 00 00 00 00 00*	RTP header
34	0x01	Burst type: A
35	0x00	Reserved
36	0x00	MFID
37	0x00	ServiceOptions: clear call
38	0x00	Privacy Algorithm ID
39	0x00	Privacy Key Id
40	0x00 00 00 00	Privacy IV
44	0x00 f8 01 a9 9f 8c e0 be 00 6a 67 e3 38 2f 80 1a 99 f8 ce 08	AMBEFrames
64	0x01 a8	Raw RSSI value
66	0x04	Current / Accepted Wireline Protocol Version
67	0x04	Oldest Wireline Protocol Version
68	Wireline Authentication ID	Uint32
72	Wireline Authentication Signation	Uint8 * 10

Table 18: Example WL_VC_VOICE_BURST Message

For the remaining WL_VC_VOICE_BURST message, most of the settings are the same as the first one:

- The CallID has to be the same as the first one.

- The slotNumber has to be the same as the first one.
- The second byte in the RTP header shall be 0x5d instead of 0xdd
- The burst type shall follow the sequence of A-F

3.5.7 Example Voice Call Message Sequences

3.5.7.1 Voice Call in Single Site Repeater System

The following figures show the message sequence that the third party application receives a voice call initiated by a radio.

In this call, there is only one call stream from the radio to the third party application. At the Call Control interface, the repeater peer uses the Call ID to identify the call stream.

The repeater has two slots which are independent. Two calls can be simultaneously supported using the two slots.

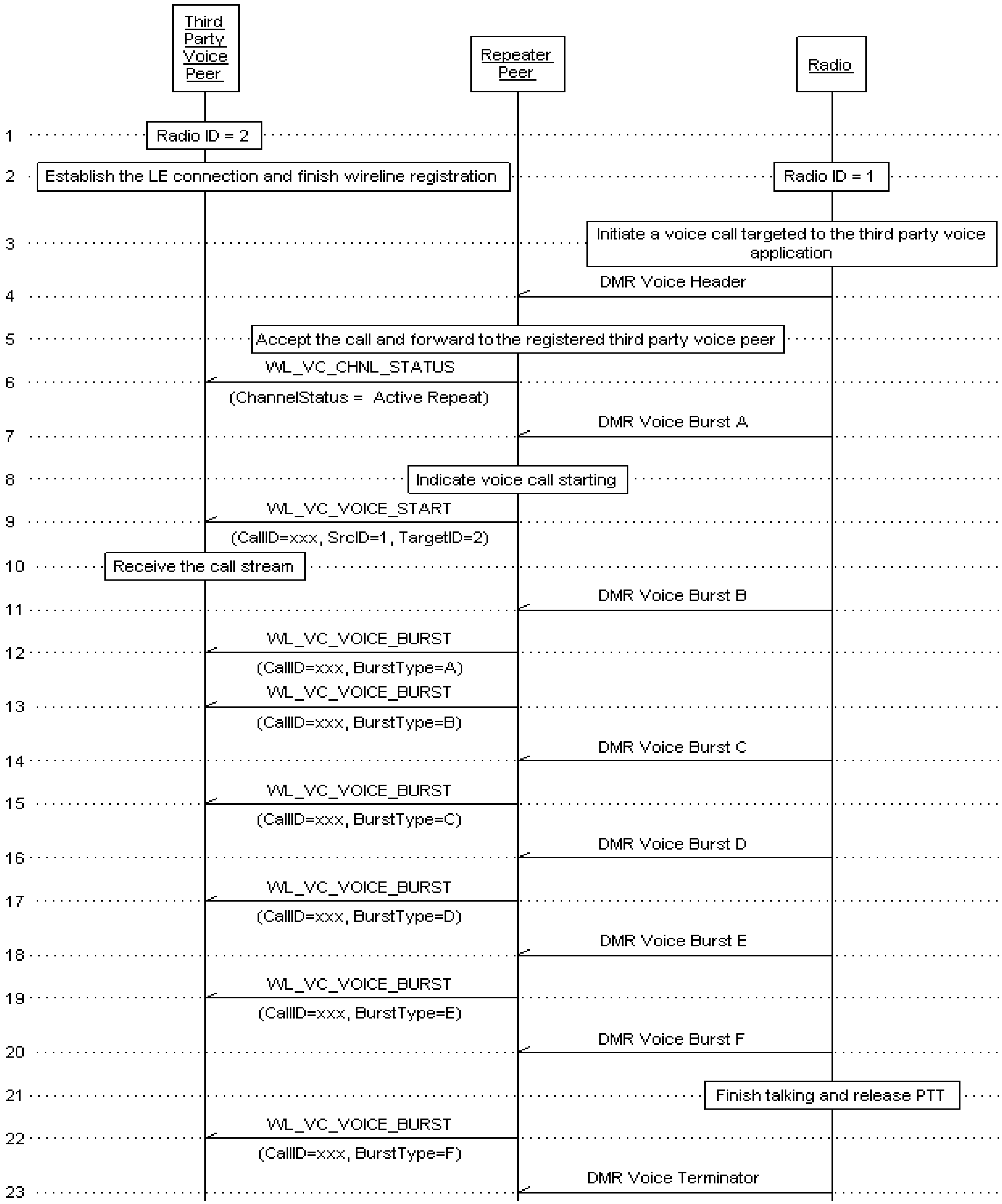


Figure 40: Application Receive Voice Call at Single Site Repeater System

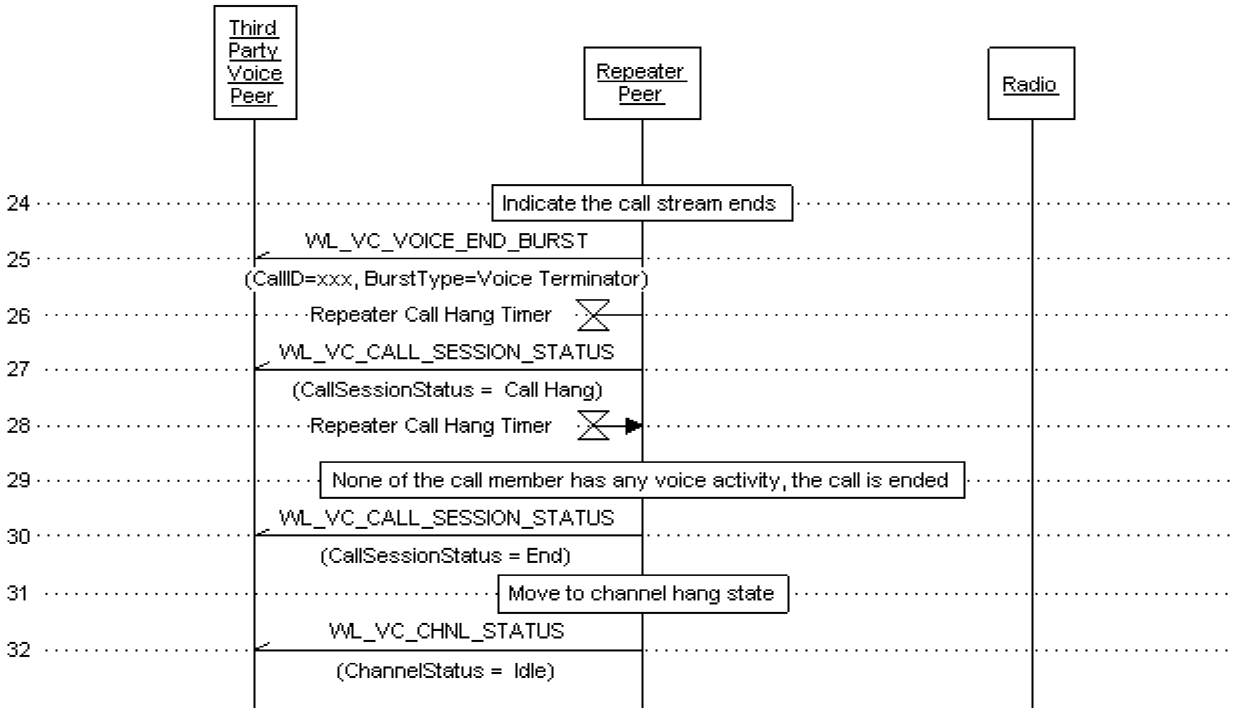


Figure 41: Application Receive Voice Call at Single Site Repeater System (Continue)

3.5.7.2 Voice Call in IP Site Connect Repeater System

The following figures show the message sequence that the third party application initiates a voice call with failure at the first try, and the radio talks back.

In the IP Site Connect system the slot can be either wide area or local area. If the call is a local area call at certain site, the third party application has to send the WL_VC_CHNL_CTRL_REQUEST to the specific repeater peer at that site and at that slot.

If the call is a wide area call, the third party application can send the WL_VC_CHNL_CTRL_REQUEST to any of the repeater peer which has the wide area channel. This applies for both call initiation and talking back.

The repeater peer is responsible for the call setup in the system on behalf of the third party application. If the call is not successfully granted as indicated by the WL_VC_CHNL_CTRL_STATUS message, the third party application shall re-send the WL_VC_CHNL_CTRL_REQUEST with a different Call ID after the channel becomes idle. It is optional for the third party application to subscribe the WL_VC_CHNL_STATUS message during the Wireline Registration, Using the WL_VC_CHNL_STATUS improves the call successful rate.

The third party application can initiate two calls simultaneously at the two slots. For example, there are 3 repeater peers in one IP Site Connect system, each of the peer has slot 1 used as the wide area channel, and slot 2 as the local area channel. At maximum, the third party application can simultaneously initiate four calls: one at the wide area channel, and three at each of the local area channel.

In the example below, after the third party application successfully initiates the call, there are two call streams: one is originated from the third party application; the other is from the radio which talks back. At the Call Control interface, two Call IDs are used to identify the two call streams. Each peer transmitting the call stream maintains its own list of the Call ID.

When a call stream ends, if the third party application wants to talk back, it shall wait for the WL_VC_CALL_SESSION_STATUS with status of Call Hang before sending WL_VC_CHNL_CTRL_REQUEST.

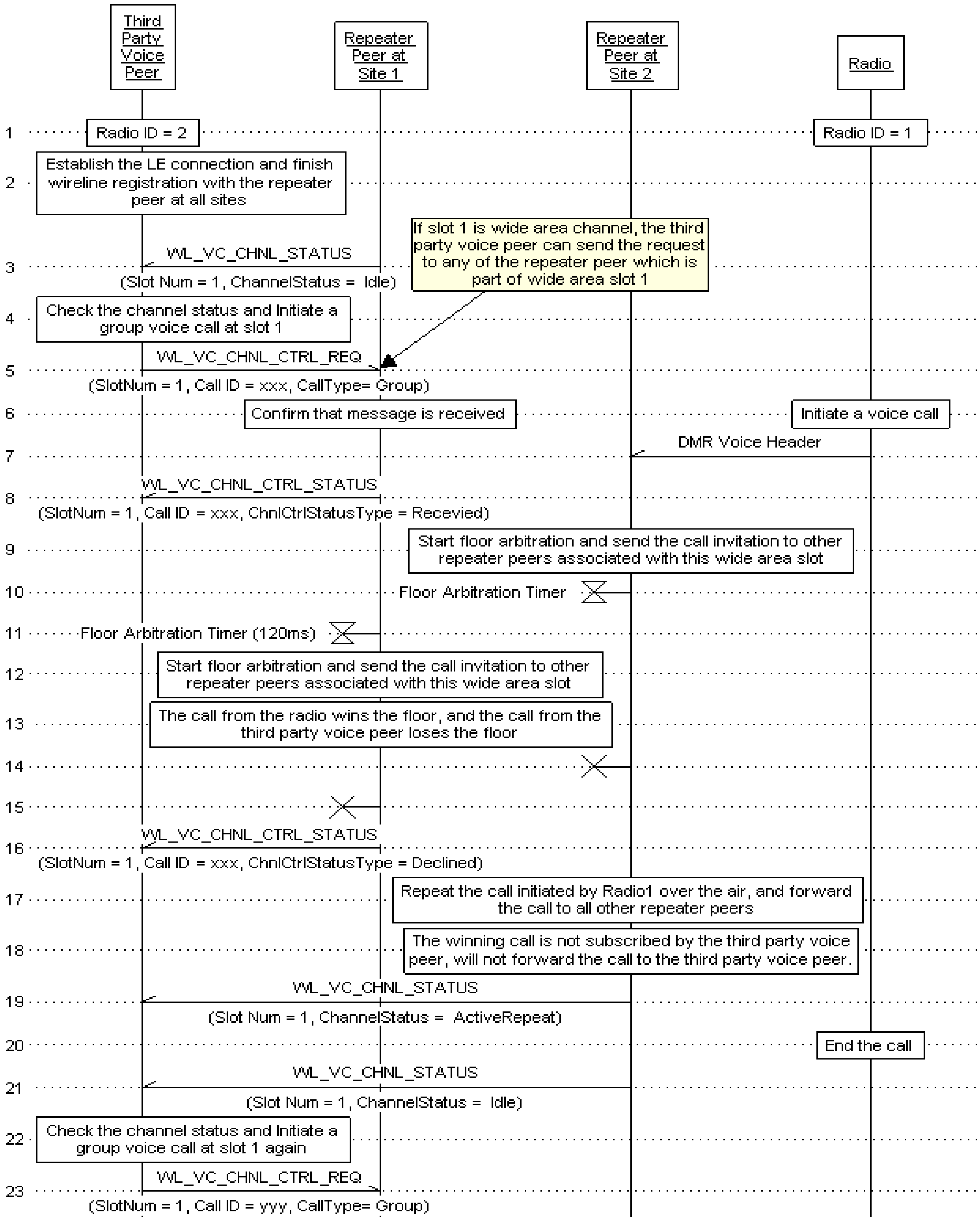


Figure 42: Application Initiate Voice Call at IP Site Connect Repeater System and Radio Talk Back

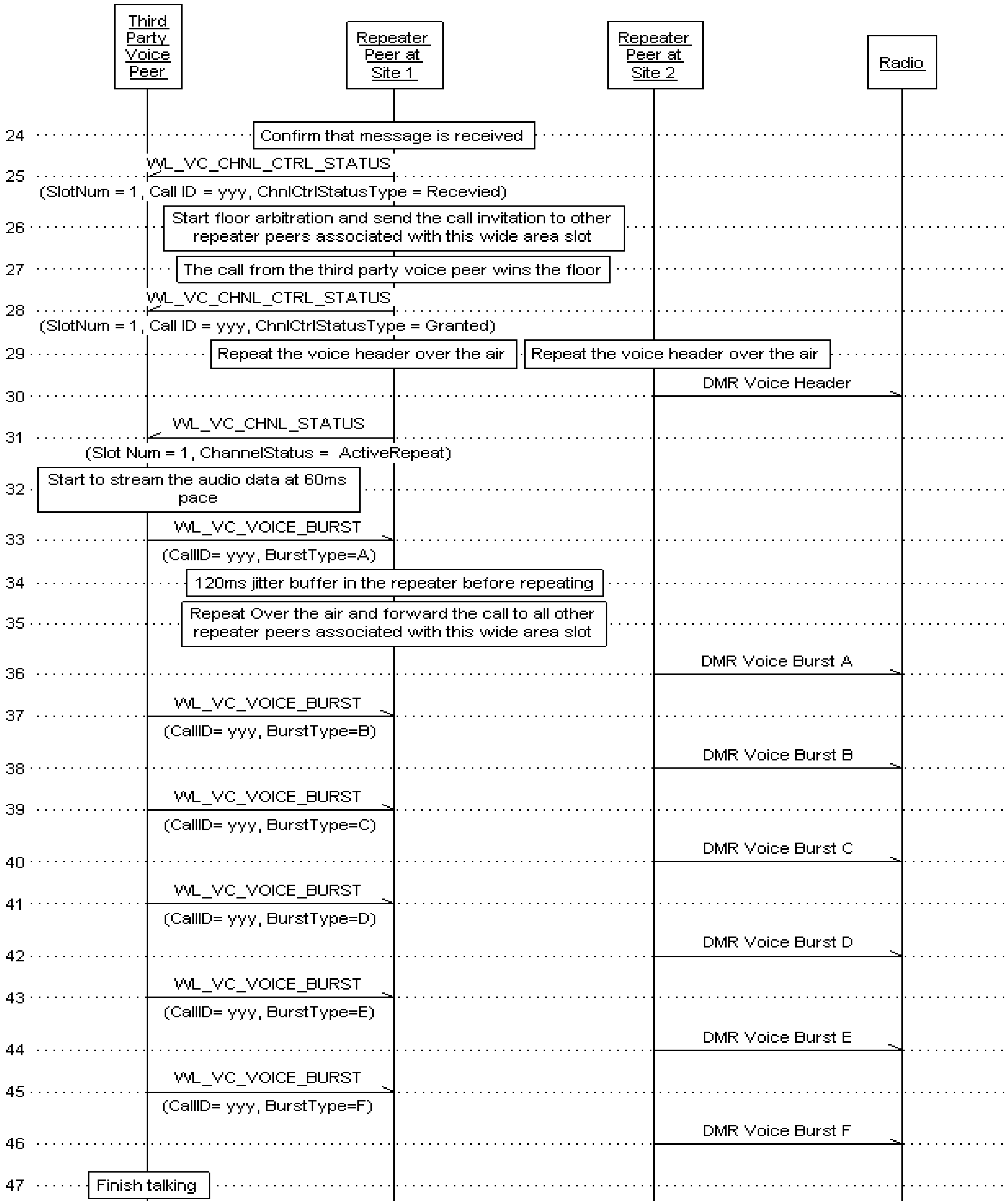


Figure 43: Application Initiate Voice Call at IP Site Connect Repeater System and Radio Talk Back (Cont)

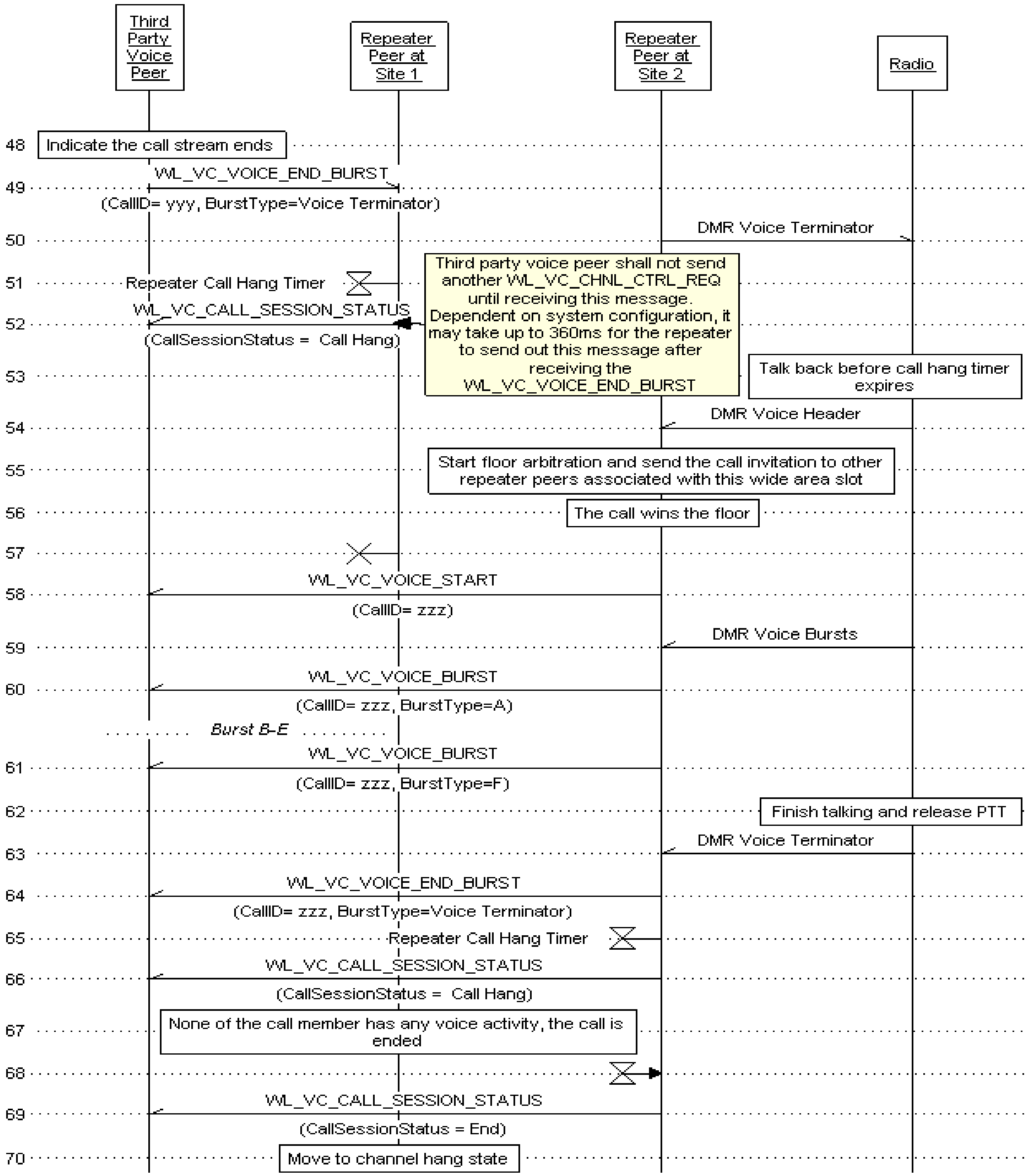


Figure 44: Application Initiate Voice Call at IP Site Connect Repeater System and Radio Talk Back (Cont)

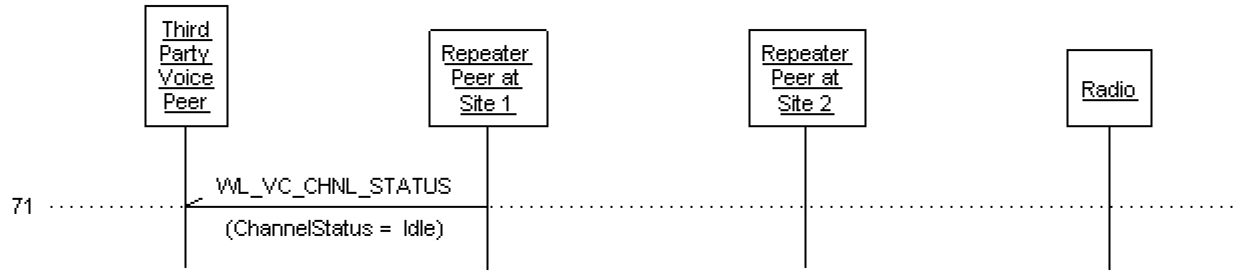


Figure 45: Application Initiate Voice Call at IP Site Connect Repeater System and Radio Talk Back (Cont)

3.5.7.3 Voice Call in Linked Capacity Plus and Capacity Plus System

The call setup procedure is identical for the trunking system at the Call Control interface. Therefore in this document unless specifically pointing out, we will treat the Capacity Plus system as a Linked Capacity Plus system with a single site. The Site Peer or Virtual Peer in the Capacity Plus system is the same as the Site Peer in the Linked Capacity Plus system.

The Rest Channel repeater has one of its slots used as the Rest Channel and acting as the Site Peer. The third party application does not know which slot of the Rest Channel Repeater is used as the Rest Channel. When the third party application initiates a call, it shall always send the WL_VC_CHNL_CTRL_REQUEST to a Site Peer, but does not need to specify the slot number. The repeater peer acting as the Site Peer handles the call setup on behalf of the third party application. Once the call is granted, the repeater peer selects the new Rest Channel (it could be the other slot in this repeater), and notifies the third party application which slot at which repeater will be used for this call with the WL_VC_CHNL_CTRL_STATUS (Status = Granted):

- The source IP address and source UDP port in the WL_VC_CHNL_CTRL_STATUS message tells which repeater is allocated for the call. This is the WAN IP address/UDP port of the repeater peer that was acting as the Site Peer at that site when receiving the WL_VC_CHNL_CTRL_REQUEST message. Also based on the Peer ID in the WL_VC_CHNL_CTRL_STATUS message, the third party application can find the WAN IP address/UDP port from the Master Peer's system map.

- The SlotNumber in the WL_VC_CHNL_CTRL_STATUS tells which slot of the repeater is allocated for the call.

The third party application shall use the allocated repeater and slot to stream the audio, including talkback during the call. The Site Peer IP address/UDP port shall only be used when a new call is initiated.

In Capacity Plus system, all the calls are within the site. The third party application does not need to track where the radio is. However, in Linked Capacity Plus system, knowing where is the radio helps to efficiently set up the call. The third party application can send the WL_VC_CHNL_CTRL_REQUEST directly to the Site Peer where the radio currently locates and specify "Do not Forward to Remote Sites" in the CallAttribute field so that the call is set up locally without involving the remote sites.

There are multiple ways to track which site the radio is currently at.

- DDMS Watcher Interface: In the MOTOTRBO CPS, when the ARS field in the LCP channel is selected as "On System/Site Change", the radio sends ARS registration message whenever it changes site. The mobility record in the DDMS is updated and the third party application is notified if it subscribess the mobility change at the DDMS Watcher interface. This is the recommended way for the third party application to get the mobility information.

- Repeater Call Monitoring Interface: The third party application can get the mobility information from the Call Transmission Status messages, which contains all the information about a call: channel, site, source ID, target ID and call status. See more details on Reference [5]. The drawback is extra network bandwidth needed to support the call logging traffic and the third party application has to parse all the call logging messages.

- Radio Check: Every time before initiating a call, the third party application initiates a Radio Check CSBK call at any Site Peer and get the mobility information based on which site the Radio Check response arrives from. The drawback is the overhead for the Radio Check CSBK call.

If for some reason, the third party application does not know where the radio is, it can still send the WL_VC_CHNL_CTRL_REQUEST to anyone of the Site Peers. The Site Peer which receives the WL_VC_CHNL_CTRL_REQUEST handles the call setup across different sites. In case the target radio is at site2 while the Site Peer receiving the WL_VC_CHNL_CTRL_REQUEST is at site1, there is one slot wasted in site1 which repeating the call over the air while the target is at site2. Also it will take up to 720ms additional time to set up the call than the case of "Do not Forward to Remote Sites".

When a call is initiated with target ID of the third party application, the call is set up without lighting up all the sites. The source repeater peer directly sends the call to the third party application since it knows the third party application's Radio ID through the Wireline Registration process.

When the third party application initiates a wide area group call, it shall only send the WL_VC_CHNL_CTRL_REQUEST to the Site Peer where this wide area group call is supported at this site. The Site Peer which receives the WL_VC_CHNL_CTRL_REQUEST handles the call setup across different sites. It uses the System Map from the Master Peer to decide which site shall be invited for this wide area group call. For example, if wide area group call 1 is associated with site 1, 2, 3, the third party application can initiate the wide area group call 1 at the Site Peer of either site 1, site 2 or site 3.

When the third party application initiates a local area group call at a site, it shall only send the WL_VC_CHNL_CTRL_REQUEST to the Site Peer where this local area group call is targeted.

The requirement that each local area group shall have a unique ID is only at the site range. Therefore, it is possible that the same local area group ID is used at different sites. Therefore in the local area group contact record, the third party application shall provide a mechanism to configure both the group ID and the site ID so that when a call is initiated with this local area group contact record, the third party application can know which site the call shall be placed. Assume local area group ID 1 is assigned to fire fighter group at site 1, and assigned to school bus driver group at site 2. Figure 46 shows the message sequence on how to initiate the local area group call with the same ID at different sites.

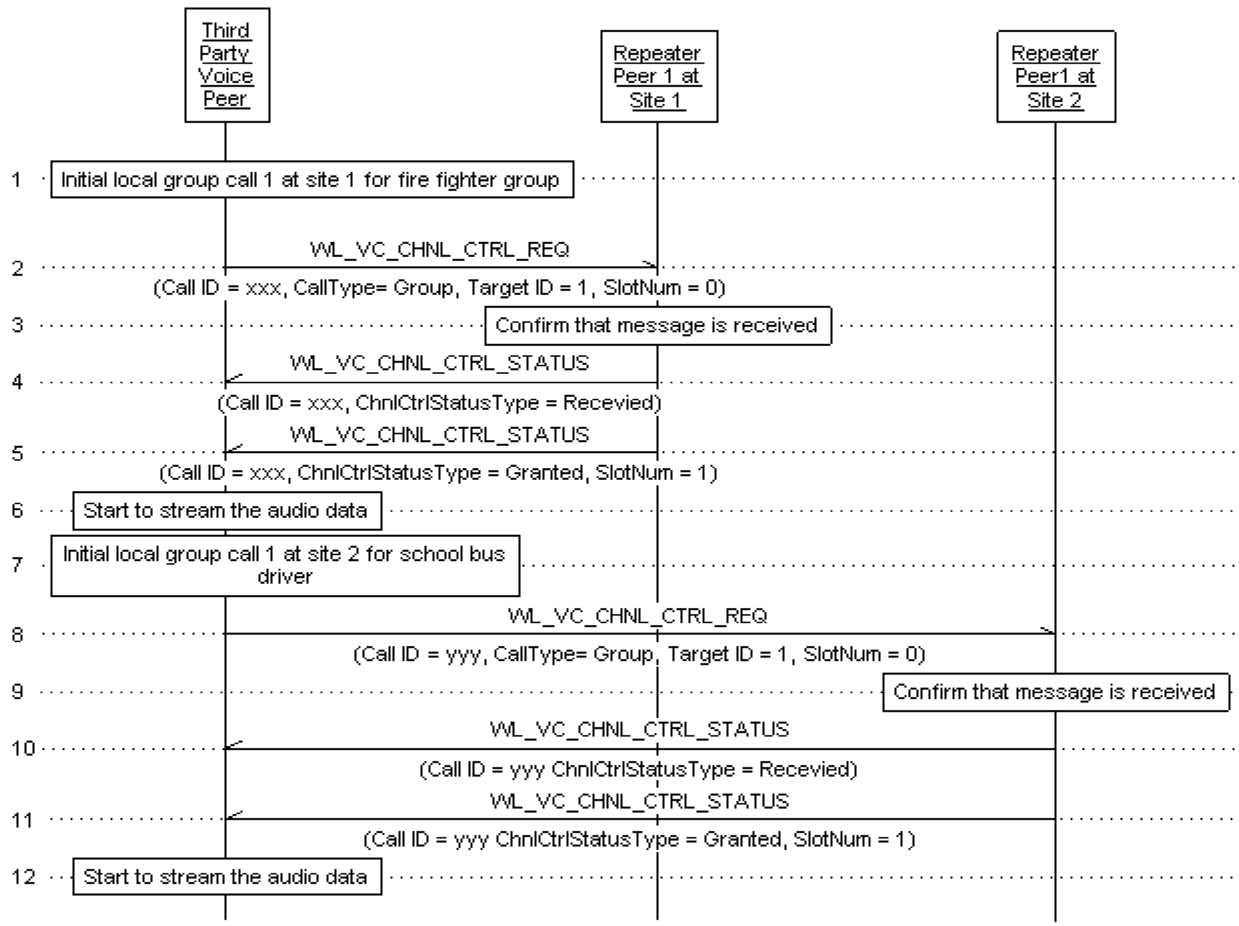


Figure 46: Local Area Group Calls at Different Sites

Unlike the conventional system, the repeater peer only sends out the WL_CHNL_STATUS to report the rest channel status when moving into or out of the rest channel busy state. The third party application only needs to check if the rest channel is available if it received the WL_CHNL_STATUS with busy rest channel status before.

Even though the third party application usually starts the voice/CSBK call with the Site Peer, which has the rest channel, there are certain limitations even when accessing the rest channel. Table 19 lists the channel access restriction to the third party application for all types of channels.

Channel Type	Access Restriction to Third Party Application
Idle Channel	Call is not allowed
Busy Channel	Call is not allowed if not party to the on-going call
Rest Channel	Call is not allowed if the call type is group call (including All call) and this talkgroup is currently active on a different channel.
Busy Rest Channel	Call is not allowed if the call type is group call (including All call) and the talkgroup is currently active on a different channel.

Table 19: Third Party Peer's Channel Access Restriction in Capacity Plus/ Linked Capacity Plus System

In all system modes, the third party application shall send the call request after receiving the WL_CALL_SESSION_STATUS of call hang if it wants to talk back. However, in the Capacity Plus and Linked Capacity Plus system, the third party application must send out the call request to the last source repeater which sends the last audio stream before the call hang expires. Otherwise the repeater declines the call request with reason code of 'non rest channel repeater'. It is recommended that the third party application shall also have the call hang timer running to avoid the race condition between a call request from the third party application and the expiration of call hang timer in the repeater. This race condition could cause the repeater to send a grant status message and immediately followed by a decline status message. This issue is planned to be fixed in R2.3. The application shall send the call request to the last source repeater if the call hang timer has not expired. If the call hang timer expires, the application shall send the call request to the rest channel repeater. By doing this, it can also help to deal the case that the call end session or channel status message was lost on the UDP connection.

3.5.7.3.1 Group Voice Call

The following figures show the message sequence that the third party application initiates a voice group call, and talks back after its' own dekey. In the talk back the third party application shall use the assigned repeater's WAN IP address and UDP port instead of the Site Peer's.

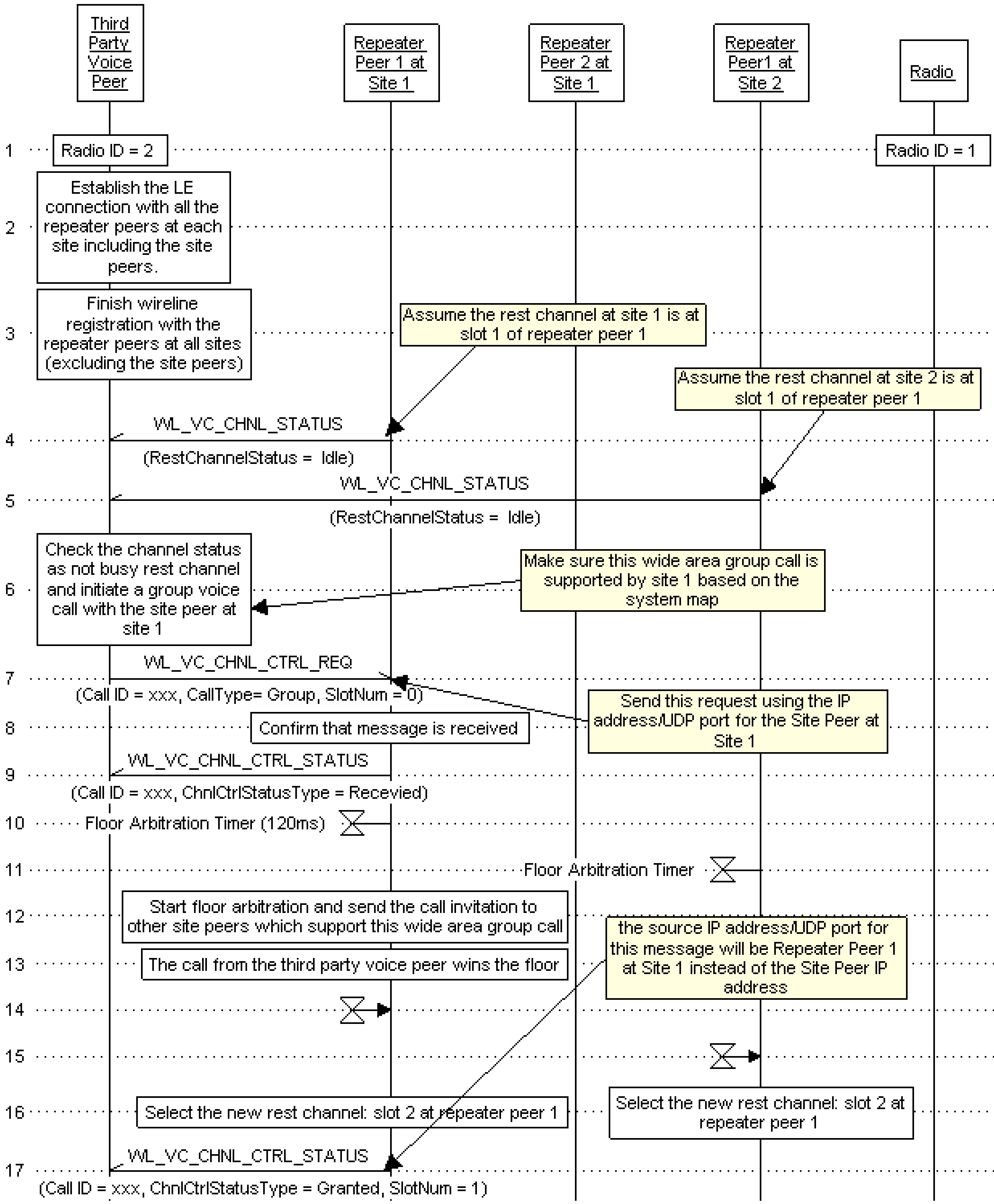


Figure 47: Application Initiate Group Voice Call at Linked Capacity Plus Repeater System

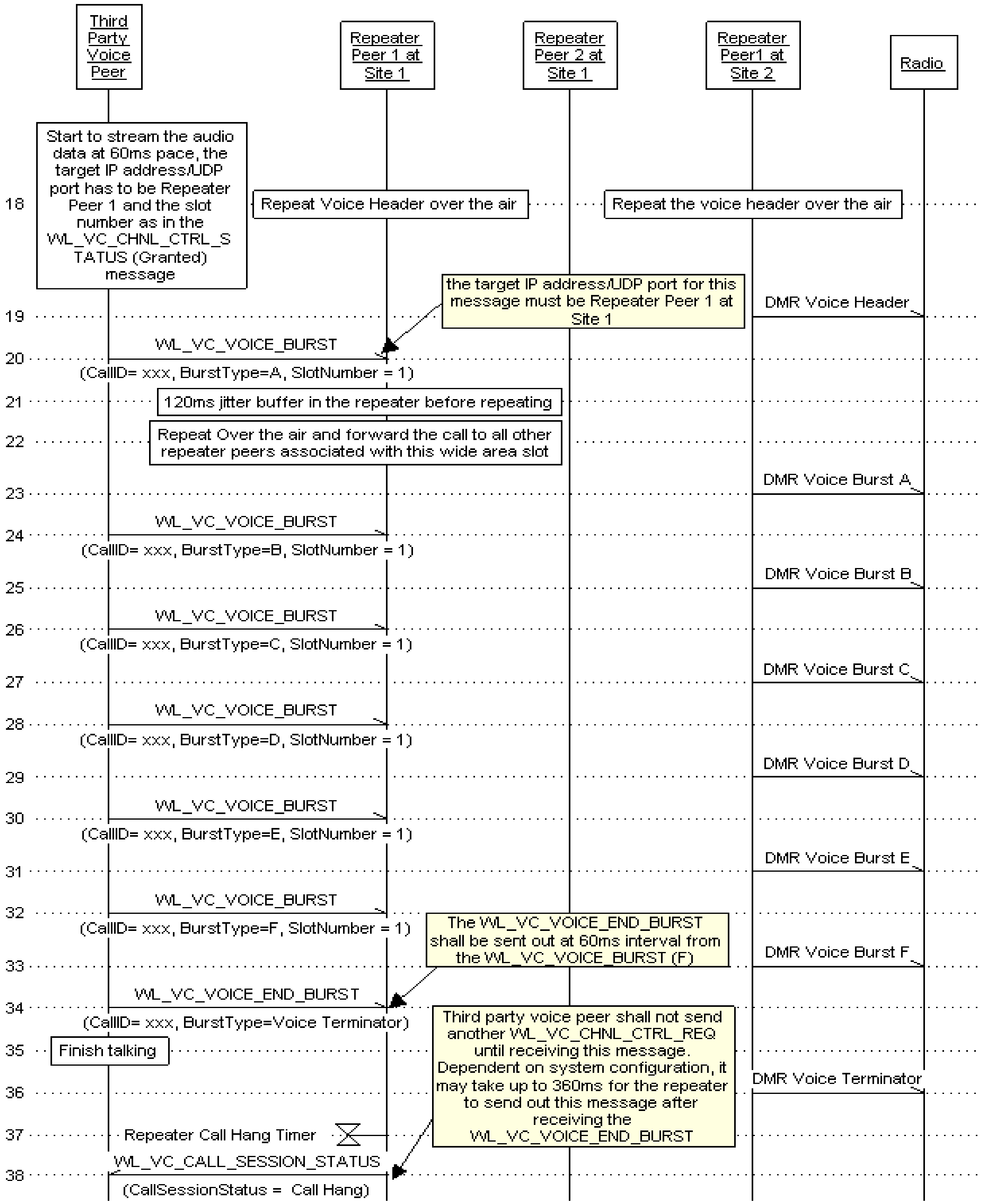


Figure 48: Application Initiate Group Voice Call at Linked Capacity Plus Repeater System (Cont)

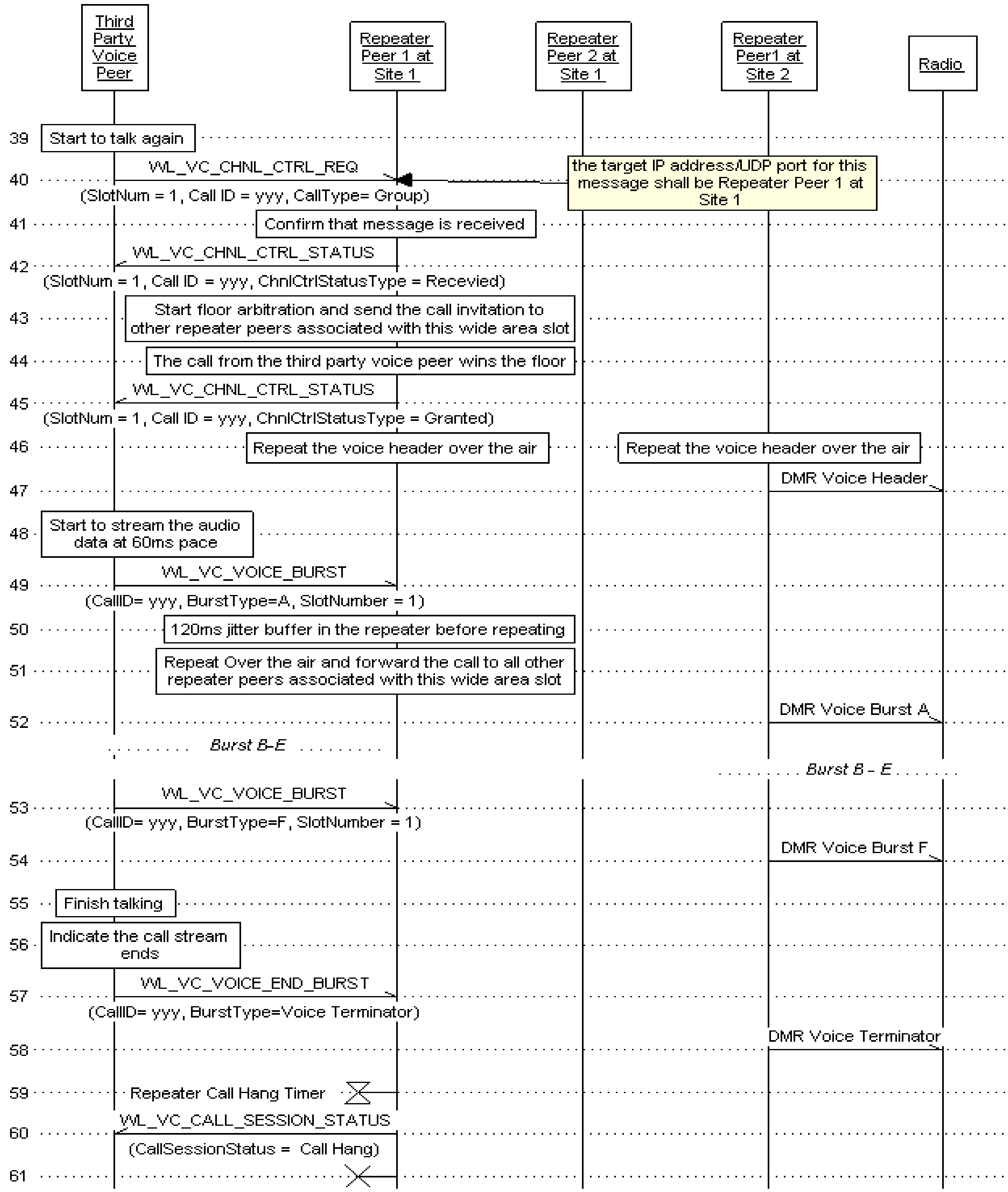


Figure 49: Application Initiate Group Voice Call at Linked Capacity Plus Repeater System (Cont)

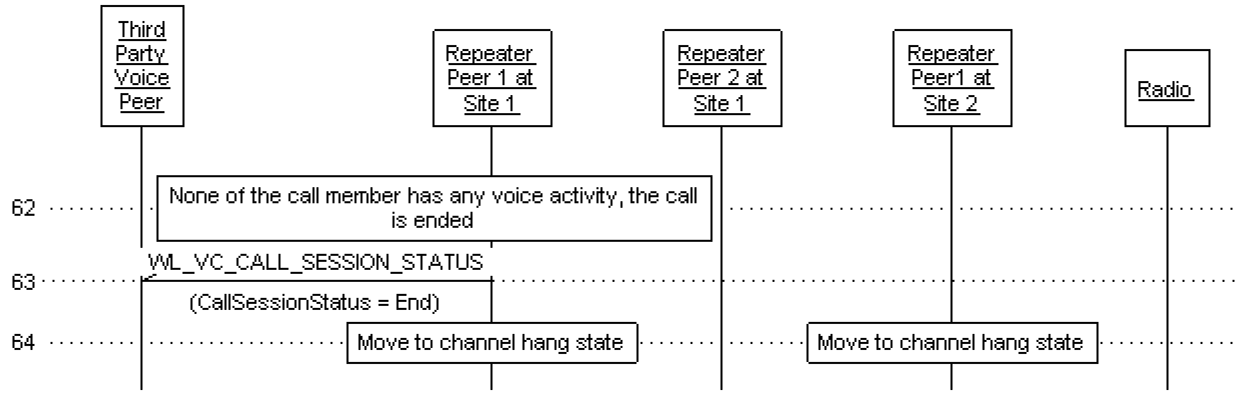
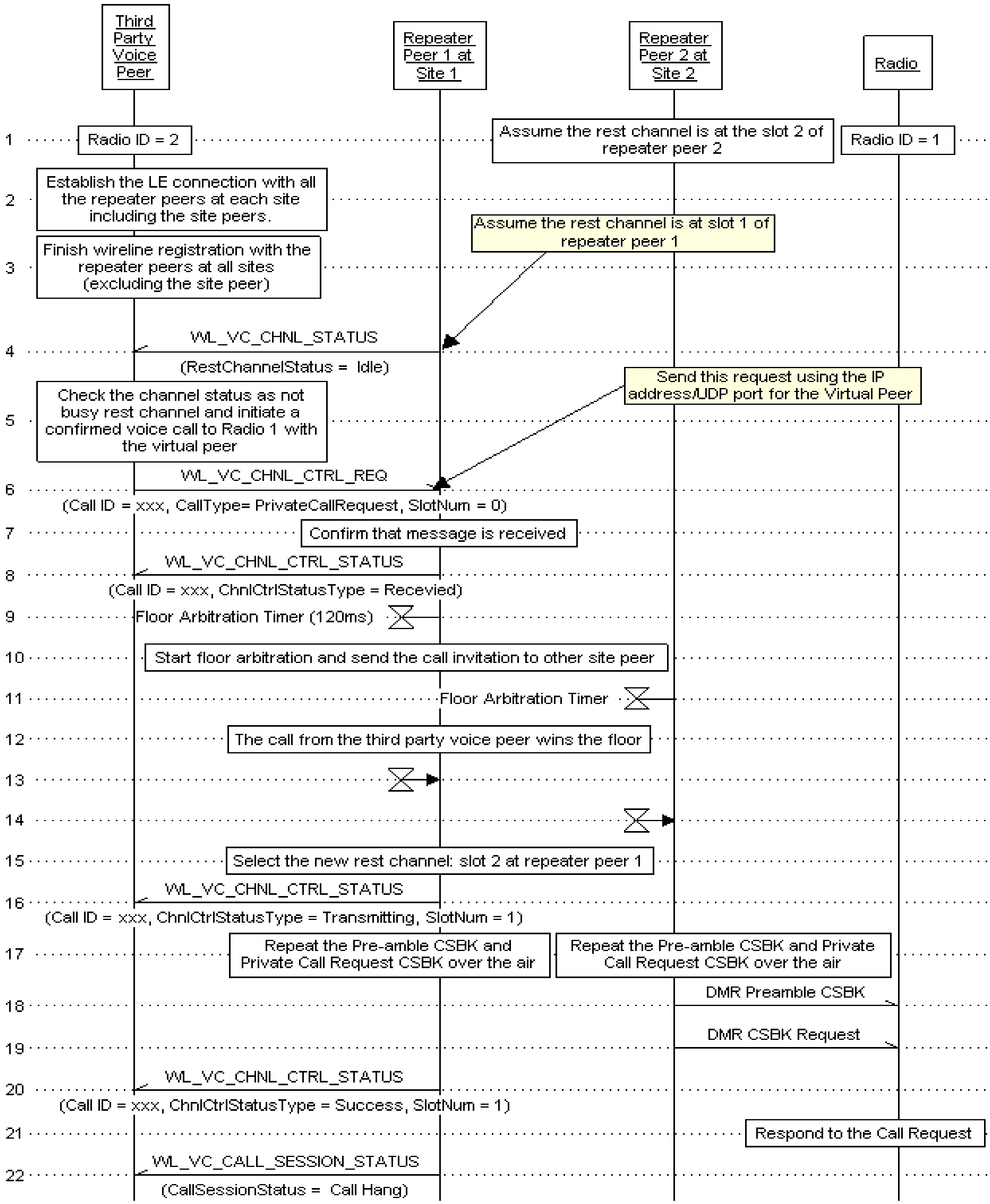


Figure 50: Application Initiate Group Voice Call at Linked Capacity Plus Repeater System (Cont)

3.5.7.3.2 Confirmed Private Voice Call

The confirmed private voice call is also called Over-the-Air Call Set Up (OACSU), which requires over the air confirmation from the target radio during the call set up. The following figures show the message sequence that the third party application initiates a confirmed private voice call to a radio. The third party application first sends the CSBK private call request to the Site Peer at Site 1. After the target radio responds, the third party application directly initiates the private voice call request to the repeater peer at Site 2 whose slot is used for the CSBK private call request transmission. The third party application shall initiate the Private Voice Call request before the voice call session changed from hang state to end state. The repeater's call hang timer duration is configurable in MOTOTRBO CPS.



1493

1494 **Figure 51: Application Initiate Confirmed Voice Call at Linked Capacity Plus Repeater System**

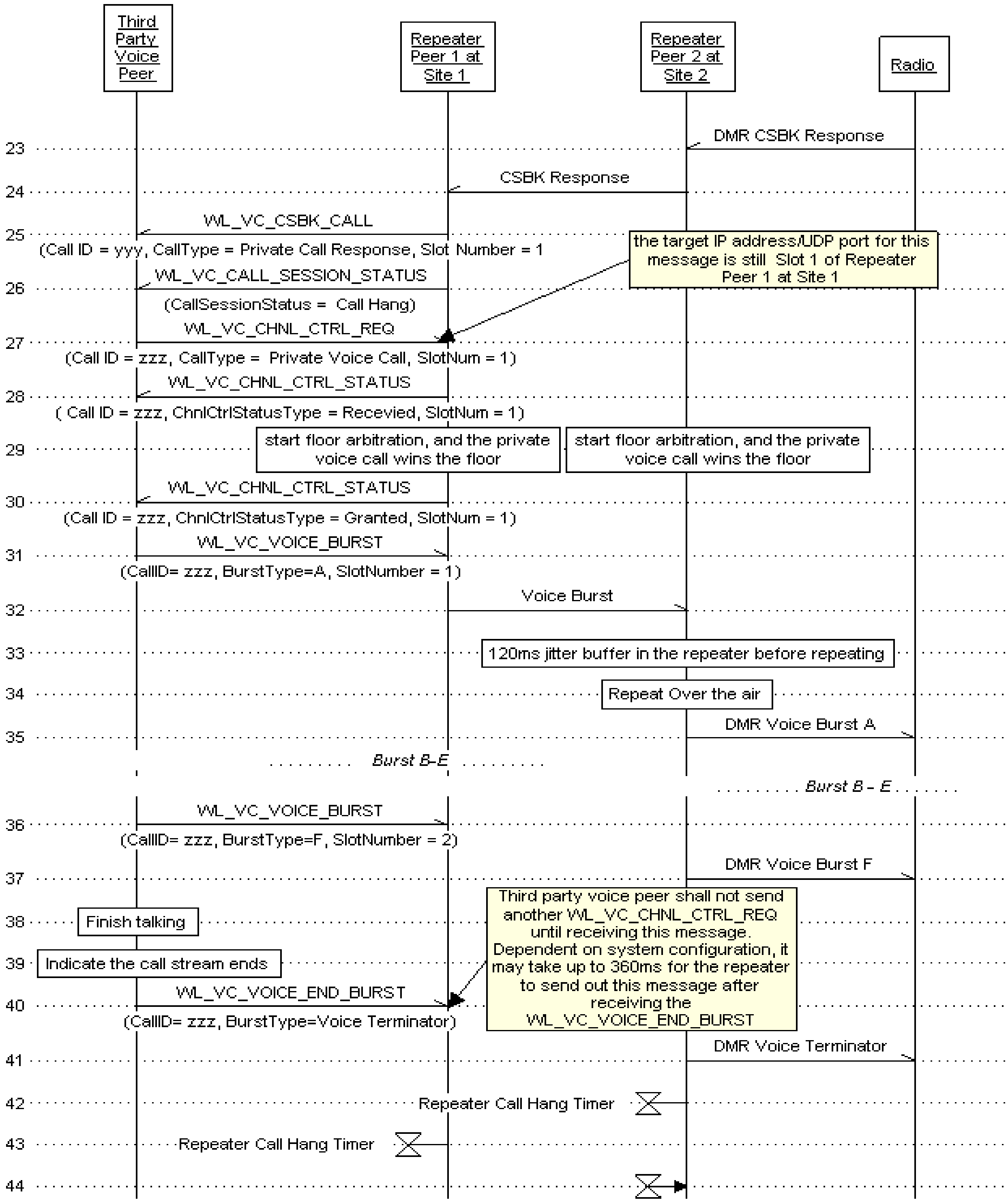


Figure 52: Application Initiate Confirmed Voice Call at Linked Capacity Plus Repeater System (Cont)

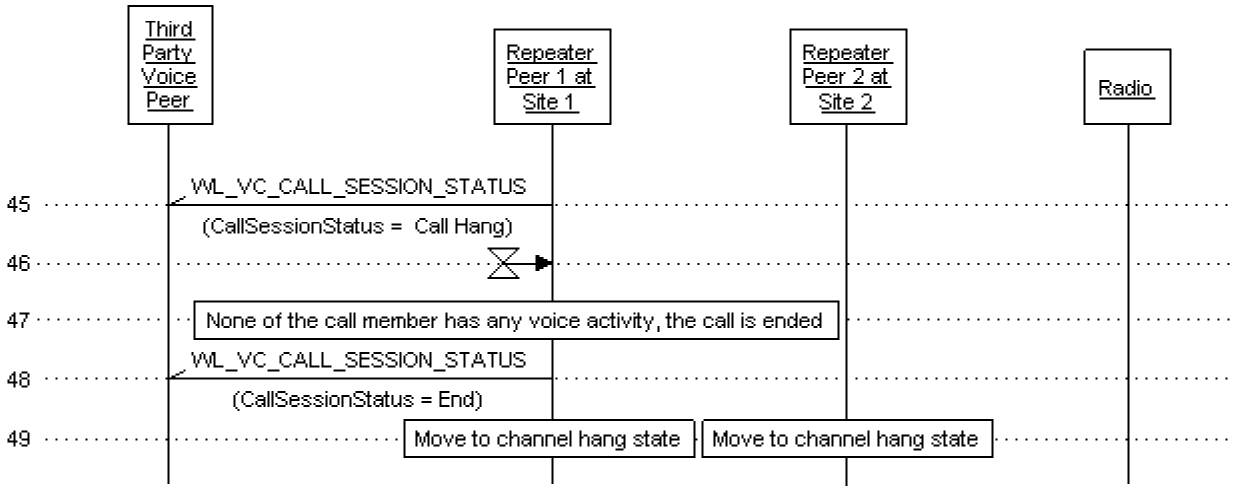


Figure 53: Application Initiate Confirmed Voice Call at Linked Capacity Plus Repeater System (Cont)

3.5.7.3.3 Handling Simultaneous New Calls

Even though both Capacity Plus and LCP system can support multiple simultaneous calls, the repeaters do not support call queuing. When more than one radios are trying to access the rest channel at the same time, their call requests are arbitrated. Only one of the radios gets the channel access and its call is set up successfully. The other radios try to access the 'new' rest channel after some hold-off delay.

A call from the third party application also gets arbitrated. Depending on arbitration outcome the third party application call may not get the access to the rest channel. The third party application is expected to retry after a minimal 60ms delay.

In LCP system, a local group call or a private call with 'do not forward' setting gets arbitrated at the local site only. A wide area group call or a private call gets arbitrated across all the LCP sites, with an arbitration window which lasts up to 120-180ms. During the arbitration new calls are not entertained. So during arbitration if a third party application call request is received, it may not get serviced and will get declined. It is expected that the third party application retries after a minimal 60ms delay. See Figure 54 and Figure 55 for example message sequence.

If the third party application has multiple calls to be initiated, it shall not initiate the second call until it receives the WL_CHNL_CTRL_STATUS with Granted or Declined status for the first call, and waits for another 60ms. The repeater peer sending the WL_CHNL_CTRL_STATUS needs the 60ms to finish the rest channel movement. If the third party application sends the WL_VC_CHNL_CTRL_REQUEST before the rest channel movement finishes, the repeater peer will sends a WL_VC_CHNL_CTRL_STATUS with failure code of NON_REST_CHANNEL_REPEATER. See Figure 54 and Figure 55 for example message sequence.

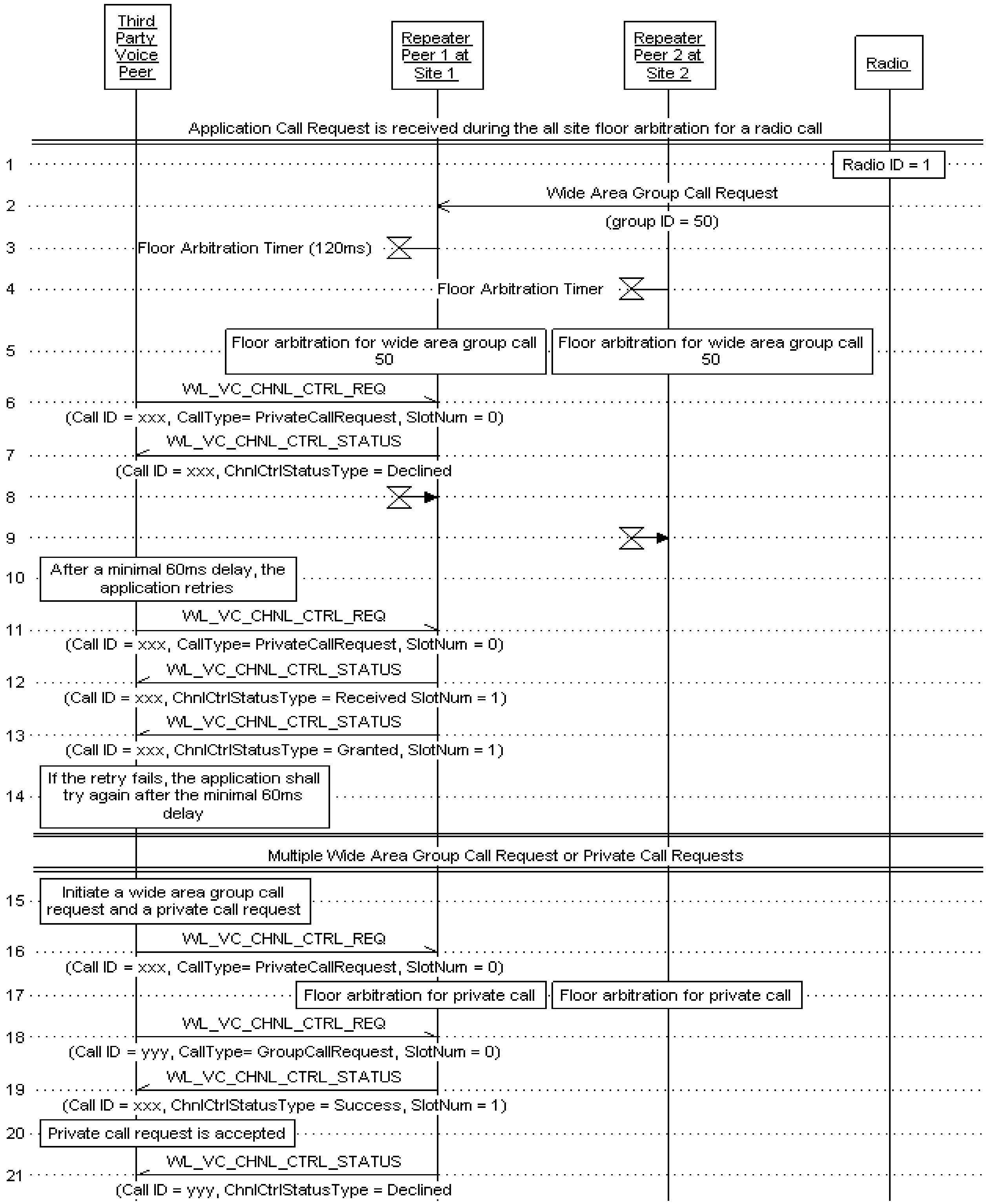


Figure 54: Call Request Received During Floor Arbitration

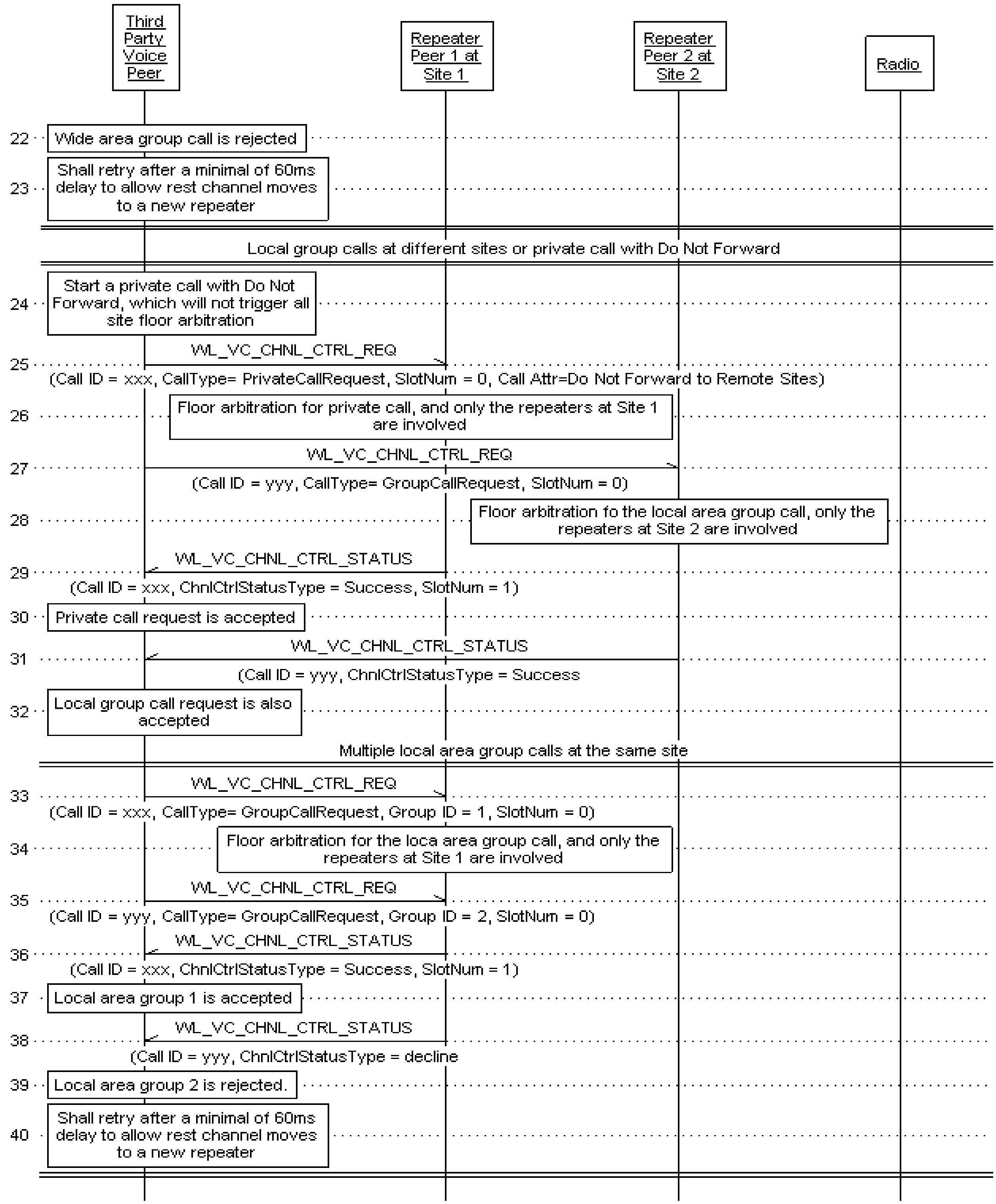


Figure 55: Call Request Received During Floor Arbitration (Cont)

It is recommended that the third party application shall automatically retry when receiving the following decline reason codes for a new call set up:

- Non-Rest channel Repeater
- Destination Slot Busy
- Race Condition Failure
- Undefined Call Failure

The third party application shall keep retrying until exhausting the third party application pre-defined number of retries or receiving the WL_VC_CHNL_CTRL_STATUS with the reason codes indicating the channel is not available, e.g. All Channel Busy, Local Group Call Not Allowed, or Destination Site Busy indication. When re-trying, the third party application should wait for a minimum of 60ms to allow the rest channel assigned to another channel. A random delay of 60ms – 420ms between the retry is recommended.

It is important to note that the access collisions could occur due to other devices (Radios, MNIS, etc.) accessing the system at the same time as the third party voice application are accessing. So both the first access and subsequent retries could also experience access collisions. Ensuring 3 or more auto retires gives a good opportunity to get the channel access.

3.5.8 Timing at Network Interface

Once starting to transmit a call over the air, the MOTOTRBO repeater peer transmits the voice bursts every 60 ms over the air. The third party application that initiates the call is expected to send the WL_VC_VOICE_BURST message every 60ms on time. Considering the arrival time difference on the network, there is a 60ms jitter buffer in the MOTOTRBO repeater peer so that the repeater peer starts the over-the-air transmission after it receives the first WL_VC_VOICE_BURST messages in a call.

In a voice call, the repeater peer sends erasure voice frame when the burst is not received before the scheduled time from the network interface. The repeater terminates the voice call when 12 continuous bursts are late.

In the diagram below, Burst2 is expected to arrive at the time of 60ms. Because of the 60ms jitter buffer in the repeater, the repeater still accepts the burst2 as long as it arrives before the time of 120ms. Otherwise the repeater substitutes it with the previous repeated burst, which is Burst1 in the diagram.

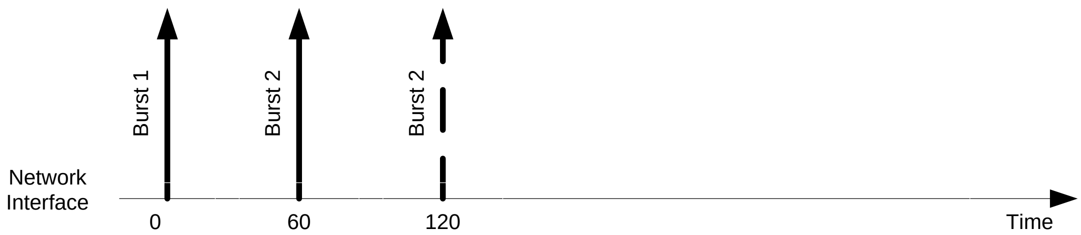


Figure 56: Burst Lost at Network Interface

3.5.9 Exception Handling

3.5.9.1 Lost Voice Packet

At the time of repeating a burst over the air, if the MOTOTRBO repeater does not receive the message containing the burst, it sends the erasure voice frame over the air.

The third party application can end a call session when it continuously misses 12 WL_VC_VOICE_BURST message from the repeater peer. The third party application does not need to send any message to the repeater peers when it ends the call session.

3.5.9.2 Late Entry Peer

There are two types of late entry: late subscriber and late peer. Late subscriber means a subscriber joins a call after the call is started. Late peer means a third party application joins a call after the call is started.

Voice calls transmitted over the air consist of at least one Voice LC Header, many voice Bursts A-F, and a Terminator with LC. The Voice LC Header and Terminator with LC carry the LC message, data frame sync, and color code while voice Burst A carries only a voice frame sync and voice Bursts B-F carry color code and embedded LC. When for some reason the subscriber misses the Voice LC Header, it can still retrieve the LC message from the voice Bursts A-F. Therefore the late subscriber is still able to join the call.

Each WL_VC_VOICE_BURST message at the Call Control interface contains the call information. The late peer can find the source ID, target ID and other call control information from the WL_VC_VOICE_BURST message and decide whether to join the call or not.

3.5.9.3 Lost Voice Terminator

When the MOTOTRBO repeater peer detects that a voice terminator is not received for the voice call from the Call Control interface, it waits for 720ms, then synthesizes a voice terminator burst, and transmits it over the air for the duration of call hang time.

3.5.9.4 Bad Voice Burst

A bad voice burst means a voice frame has too many errors to be corrected by FEC. When the AMBE decoder in the MOTOTRBO repeater cannot correct all the bit errors in the 20ms voice frame, the Bad Voice Burst bit in the AMBE Encoded Voice Frame field of the WL_VC_VOICE_BURST message is set to 1 for the voice frame. Another case is during the voice transmit interruption, the LCP repeater peer may send the WL_VC_VOICE_BURST message with Bad Voice Burst bit being set to 1 for about 1600ms as shown in Figure 64.

In both cases, the third party application shall indicate the bad voice to the AMBE vocoder for this 20ms voice frame so that it can enable internal mitigation in the output audio: set the “AMBE_LOST_FRAME_FLAG” in the cmod field of the voice decoder function call if the AMBE software codec is used; set the “LOST_FRAME” in the DCMODE_IN field for the 20ms frame if the AMBE hardware codec is used.. The third party application shall not terminate the call session when the Bad Voice Frame is set in the WL_VC_VOICE_BURST message.

When the MOTOTRBO repeater receives a WL_VC_VOICE_BURST with Bad Voice Frame of 1, it repeats the voice frame without further processing.

3.5.9.5 Lost Response

When the third party application sends some of Call Control messages, a response message is expected from the repeater peer. Table 20 provides guidance on how long the third party application shall wait for the response after sending out the request. The third party application can re-send the request after the waiting time expires. For example, after the third party application sends the WL_CHNL_CTRL_REQ for non-Transmit Interruption case, if the third party application does not receive the WL_CHNL_CTRL_STATUS with status of received within 500ms, it can re-send the WL_CHNL_CTRL_REQ message; if it does not receive the WL_CHNL_CTRL_STATUS with status of granted or declined within 500ms after receiving the WL_CHNL_CTRL_STATUS with status of received, it can re-send the WL_CHNL_CTRL_REQ message. See the message sequence below for the detailed description. It is recommended to use different Call ID in each retry of the WL_CHNL_CTRL_REQ.

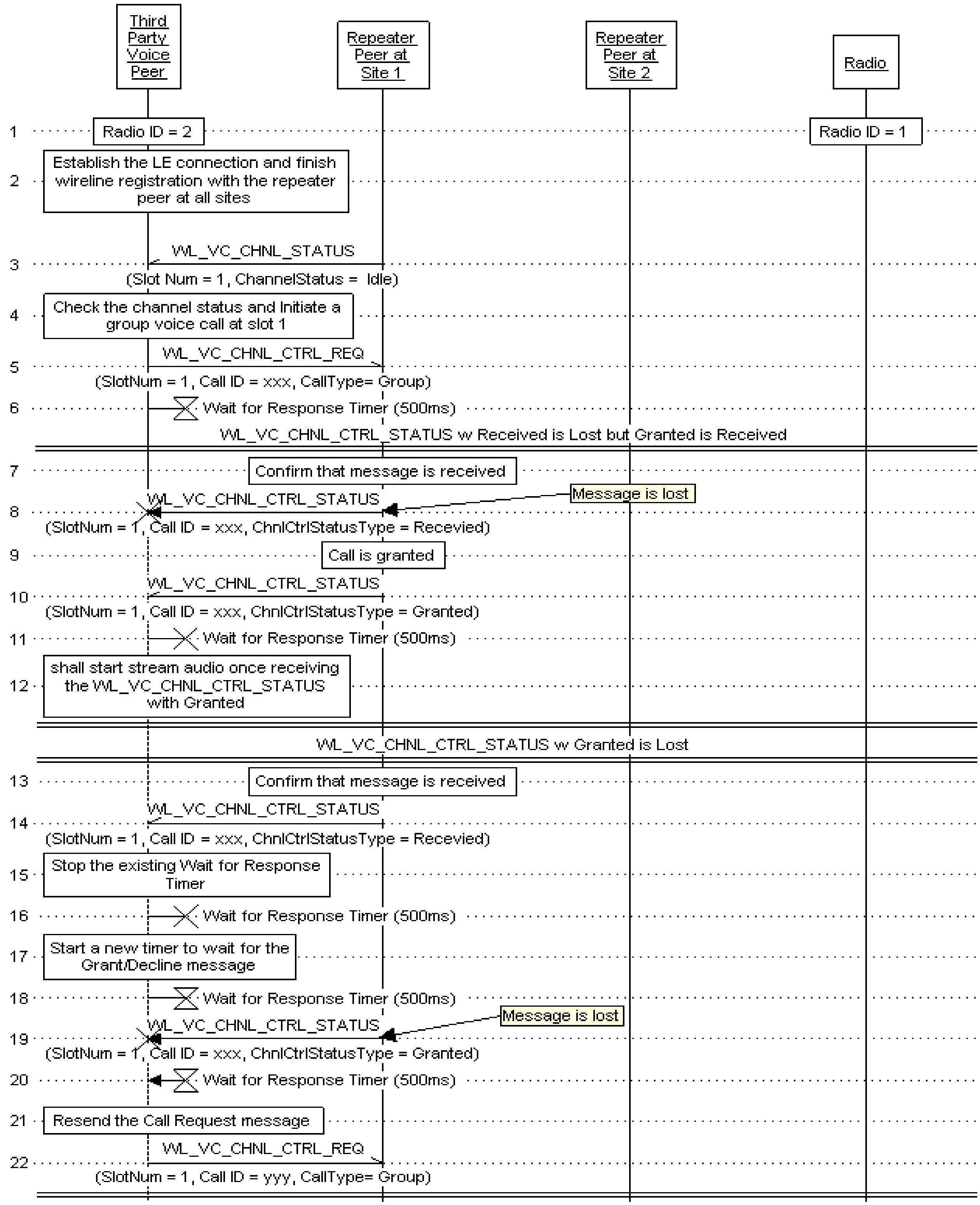


Figure 57: Application Retries When Response is Lost

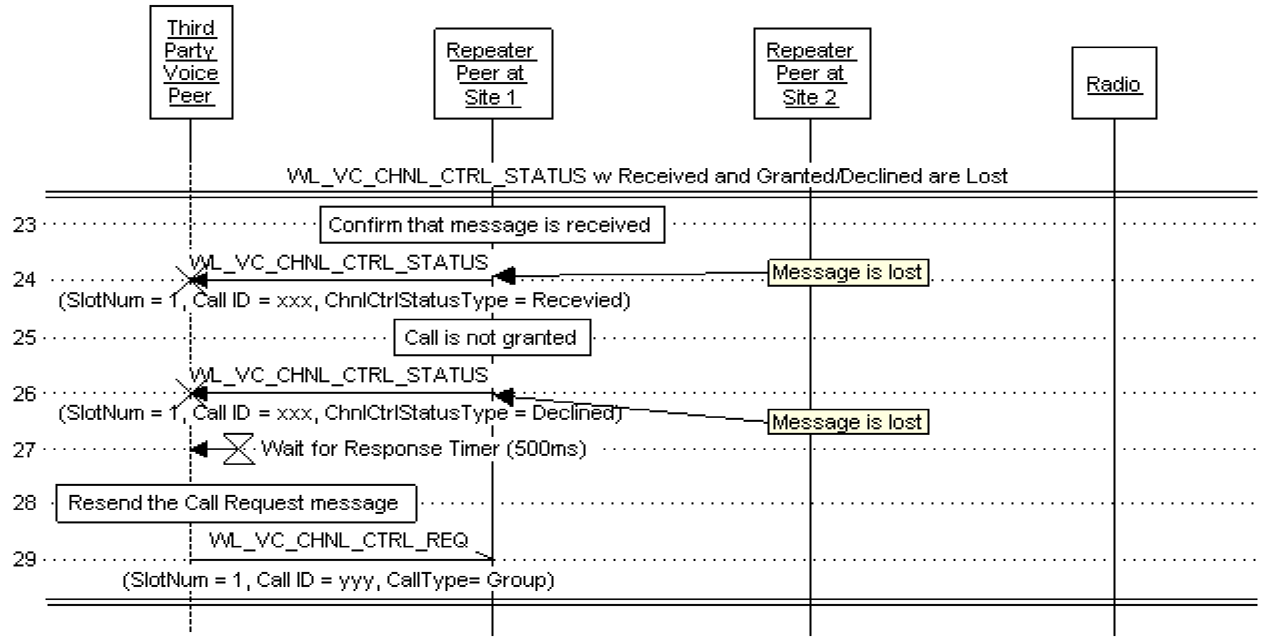


Figure 58: Application Retries When Response is Lost (Continue)

	WL_Registration_ Request	WL_CHNL_STATUS _QUERY	WL_CHNL_CTRL _REQ for non-Transmit Interruption or CSBK Call	WL_CHNL_CTRL_ REQ for Transmit Interruption
Waiting Time	**300ms**	300ms	1000ms (500ms waiting for response with status of received for both voice and CSBK call; 500ms waiting for response with granted/declined for voice call; 500ms waiting for response of transmitting/declined for CSBK call)	2500ms (500ms waiting for response with status of received; 2000ms waiting for response with granted/declined)

Table 20: Waiting Time per Request Type

The WL_CHNL_CTRL_REQ for Transmit Interruption means the access criteria in the WL_CHNL_CTRL_REQ is set to Transmit Interrupt. See section 3.5.12 for more details on transmit interruption.

3.5.9.6 Lost WL_VC_CALL_SESSION_STATUS Message

After a call stream ends, the repeater is expected to send the WL_VC_CALL_SESSION_ STATUS message to indicate it is in Call Hang state now, then the third party application can send a WL_VC_CHNL_CTRL_REQUEST to the repeater which sends the last audio stream if it wants to talk back.

The WL_VC_CALL_SESSION_ STATUS of Call Hang could be lost in the network, the third party application can assume the repeater is in the Call Hang if it does not receive WL_VC_CALL_SESSION_STATUS message after 500ms since it receives the WL_VC_VOICE_END_BURST message.

The WL_VC_CALL_SESSION_STATUS of Call Session End could also be lost in the network, the third party application shall start a call hang timer once it sends or receives the WL_VC_VOICE_END_BURST message. The call hang timer in the application shall be 60ms longer than the repeater's call hang timer. The application shall stop the call hang timer once it receives an audio stream from the radio or sends a call request for a talkback.

3.5.9.7 Process of WL_VC_CALL_SESSION_STATUS of Call Session End

The WL_VC_CALL_SESSION_STATUS of Call Session End usually indicates a voice call is over. Because internally the repeater may send the Call Session End message before the Call Hang timer expires, and the repeater will still process the Call Request from either the application or the radio before the Call Hang timer expires. Therefore, dependant on the phase of the call, the application may take different actions.

1. If the application is not sending a request for a talkback, and receive the Call Session End message, and then receive the audio stream from the radio, the application shall still process the audio stream. Only when the application's call hang timer expires, it indicates to the end user the call is over.

2. If the application has sent a call request for a talkback, and receives the Call Session End message, the application shall wait for the WL_VC_CHNL_CTRL_STATUS message.

   a) If the call request is granted, the application shall start to stream the audio.

   b) If the call request is declined, the application shall re-send the call request to the Site Peer in the Capacity Plus or Linked Capacity Plus system

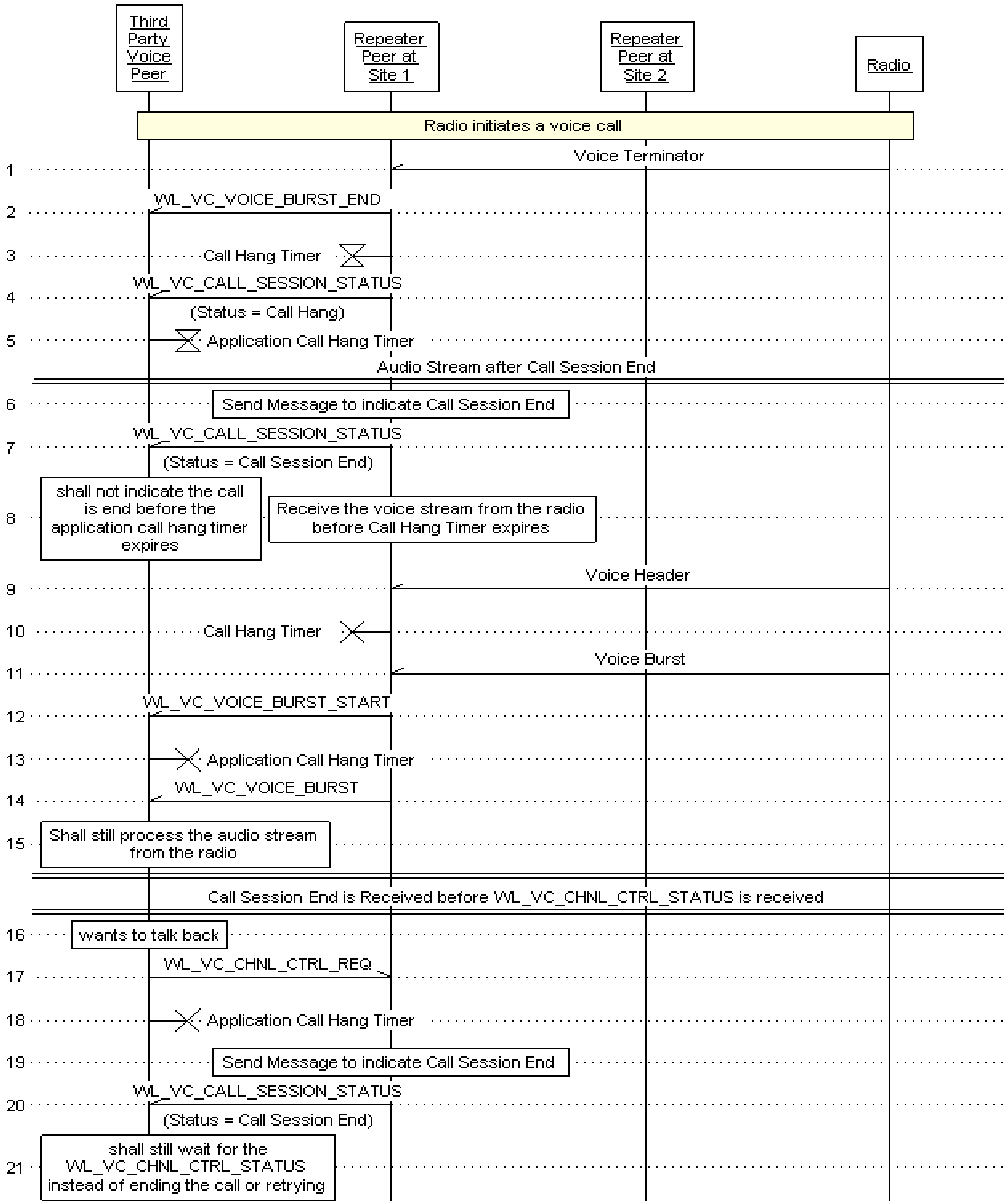


Figure 59: Process Call Session End Message

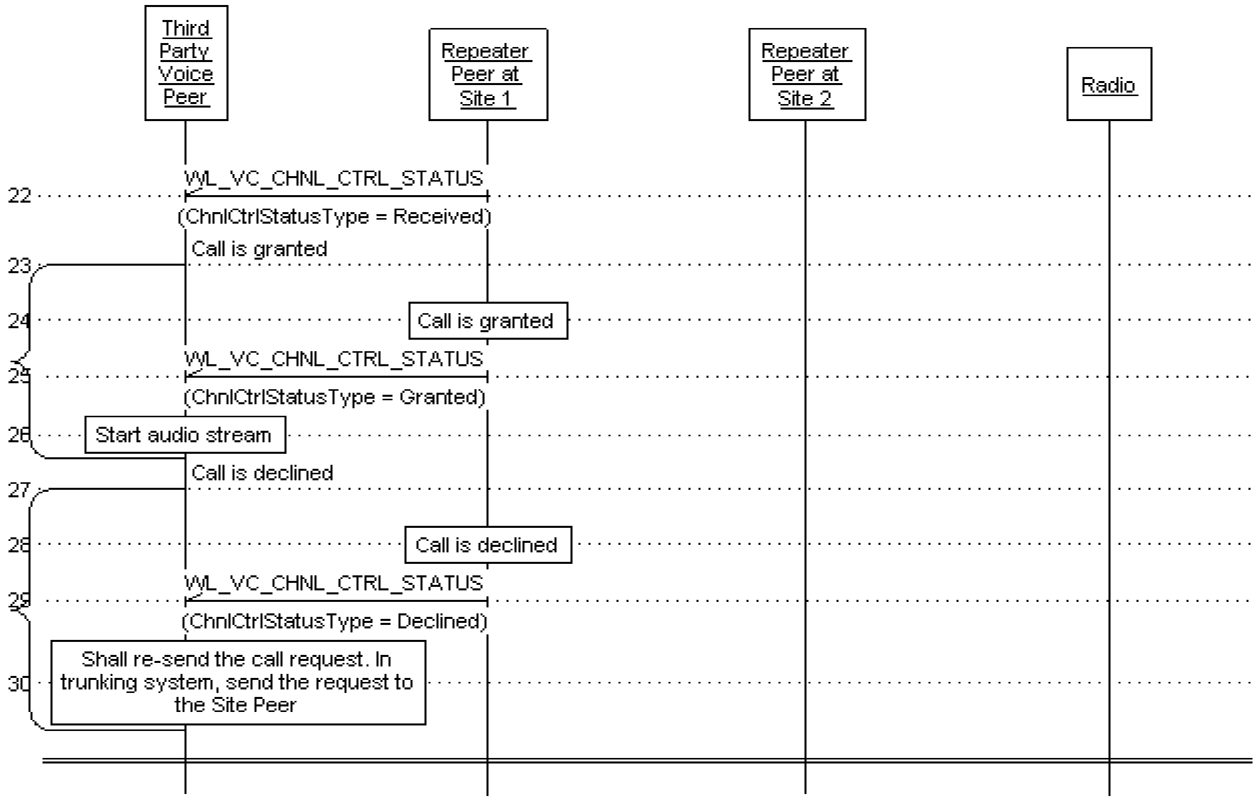


Figure 60: Process Call Session End Message (Continue)

3.5.9.8 Lost WL_VC_CHNL_STATUS

As indicated by section 3.5.7.2, the third party application can use the WL_VC_CHNL_STATUS message to know if a channel in a conventional system is busy or idle, or if there is available channel in a Capacity Plus or Linked Capacity Plus system. However, this message could also be lost in the network. In case the third party application gets a WL_VC_CHNL_STATUS indicating the channel is busy, it will wait for the WL_VC_CHNL_STATUS for the indication of idle channel before initiating a call. The third party application shall have the mechanism to get out of waiting state in case the message is lost, e.g. querying the channel status after the timer expires.

3.5.9.9 Support Maximal Number of Simultaneous Call Sessions

Through the Call Control Interface, the third party application can receive all the voice or CSBK calls no matter which channel it takes place after the third party application registers all the calls for recording purpose, or the third party application itself participates the calls at all the channels at the same time. Therefore, the third party application is expected to design and test to support the maximum number simultaneous calls that can occur in a system.

Assume only one third party application is connecting to the repeater system, Table 21 gives the maximum simultaneous call session supported per each system type.

MOTOTRBO System Type	Maximum Repeater System Size When One Third party application Connecting to the System	Maximum Number of Simultenous Call Session
IP Site Connect	**14 Sites**	14 * 2 = 28 (when all the channels are Local Area Channel)
Capacity Plus	**8 Trunked Repeater, 12 Data Revert Repeater**	6 * 2 = 16
Linked Capacity Plus	**14 Sites, each site has 8 Trunked Repeater and 4 Data Revert Repeater.**	14 * 8 * 2 = 224 (when all the calls are local group calls)

Table 21: Maximum Simultaneous Call Session per System Type

3.5.9.10 Console Audio Loop Back

In R2.3, the repeater peer will not stream the audio originated from one application to another repeater or application peer. Therefore, the audio recording application will not be able to record all the voice calls originated from the console application

3.5.10 Call Access Criteria

The third party application can access the channel with the following access criteria:

- Polite Access: only if the channel is idle
- Transmit Interrupt: interrupt the on-going interruptible call
- Remote Dekey: remotely dekey the sourcing radio which is initiating the on-going interruptible call.
- Impolite Over Voice Access: impolitely take-over the on-going call by replacing the inbound audio with its own audio while the source radio is still keying up

Table 22 shows the access criteria supported and not supported in each type of the system. The following are the term definitions in the table:

- In-Call means the third party application call being initiated is party to the on-going call.
- Not-In-Call means the third party application call being initiated is NOT party to the on-going call.

- Supported means the access criteria will success
- Not Supported means the access criteria will not success, a decline status will be sent

System Mode	Call Type	Impolite*	Transmit Interrupt**	RVD**
Single Site / IPSC Conventional	In-Call	Supported	Supported	Not Supported
	Not-In-Call	Supported	Not Supported	Supported
Capacity Plus	In-Call	Supported	Supported	Not Supported
	Not-In-Call	Supported[1]	Not Supported	Supported[1]
Linked Capacity Plus	In-Call	Supported[3]	Supported	Not Supported
	Not-In-Call	Not Supported	Not Supported	Supported[1,2]

Table 22: Access Criteria per System Type

*Supported when the call being taken over is a voice call from a radio. Not supported if the call is a data or CSBK call, or a voice call from another third party application,

**Supported when the call being taken over is an interruptible voice call from a radio. Not supported if the call is a data or CSBK call, or a voice call from another third party application.

[1]The access is supported on a busy rest channel

[2]The access is not supported during call hang

[3]The third party application must send the impolite voice call request to the repeater where the on-going call is originating or the repeater sourcing the on-going call to the third party application.

The third party application initiates an emergency alarm or an emergency voice call. The emergency call can take over any on-going voice call in any system mode. However, in SS/IPSC/CPC mode the emergency call cannot take over a Data or a CSBK call. If the repeater is unable to grant access to the emergency call, it sends a decline response to the third party application.

The impolite access criteria allow the third party application to take over an on-going voice call. When accessing an idle channel the third party application could send the WL_VC_CHNL_CTRL_REQUEST with impolite access so that the third party application's call has a higher priority during call arbitration. While this approach makes a console call win during arbitration, it may not always turn out in this manner. If the WL_VC_CTRL_REQUEST with impolite access is received during repeater's floor arbitration phase for a call initiated by a radio, the radio's call wins the channel. However, the third party application can transmit interrupt or impolite take-over the radio's call.

The Interrupt access criteria, does not assign a higher priority for the third party application call during arbitration. Using Interrupt access criteria when the channel is idle is same as using the Polite access criteria.

It should be noted that an impolite takeover only disconnects repeater's inbound to then outbound path and injects the third party application call into the outbound path. The inbound transmission from the original on-going call radio remains ON and could prevent other radios to respond to the third party application. The takeover via transmit Interrupt or remote dekey, stops the inbound transmission which then allows other radios to respond to the third party application.

Section 3.5.11 and 3.5.12 describe in details on how the third party application can interrupt or impolitely takeover an on-going call.

3.5.11 Impolite Access Over On-Going Call

To impolitely take over an on-going call, the third party application sends a WL_VC_CHNL_CTRL_REQUEST to the repeater peer. In a multisite system, the source repeater is the one which receives the call over the air from the radio. The remote repeater is the one which receives the call over the network interface. In the IPSC system, the impolite take-over request can be sent to the source repeater or one of the remote repeaters. In the LCP system, the impolite take-over request has to be sent to the source repeater. . The WL_VC_CHNL_CTRL_REQUEST shall have with the following setting:

- SlotNumber: the slot number the to-be-dekeyed call is on-going
- CallType: Voice Call
- SourceID: the Radio ID of the third party application
- TargetID: the Radio ID of the radio which is transmitting
- Access Criteria: Impolite
- Preamble Duration: 0
- CSBK Parameters: blank

Figure 61 shows the message sequence that the third party application impolitely takes over the voice call. It is not required that the third party application has to be part of the on-going call.

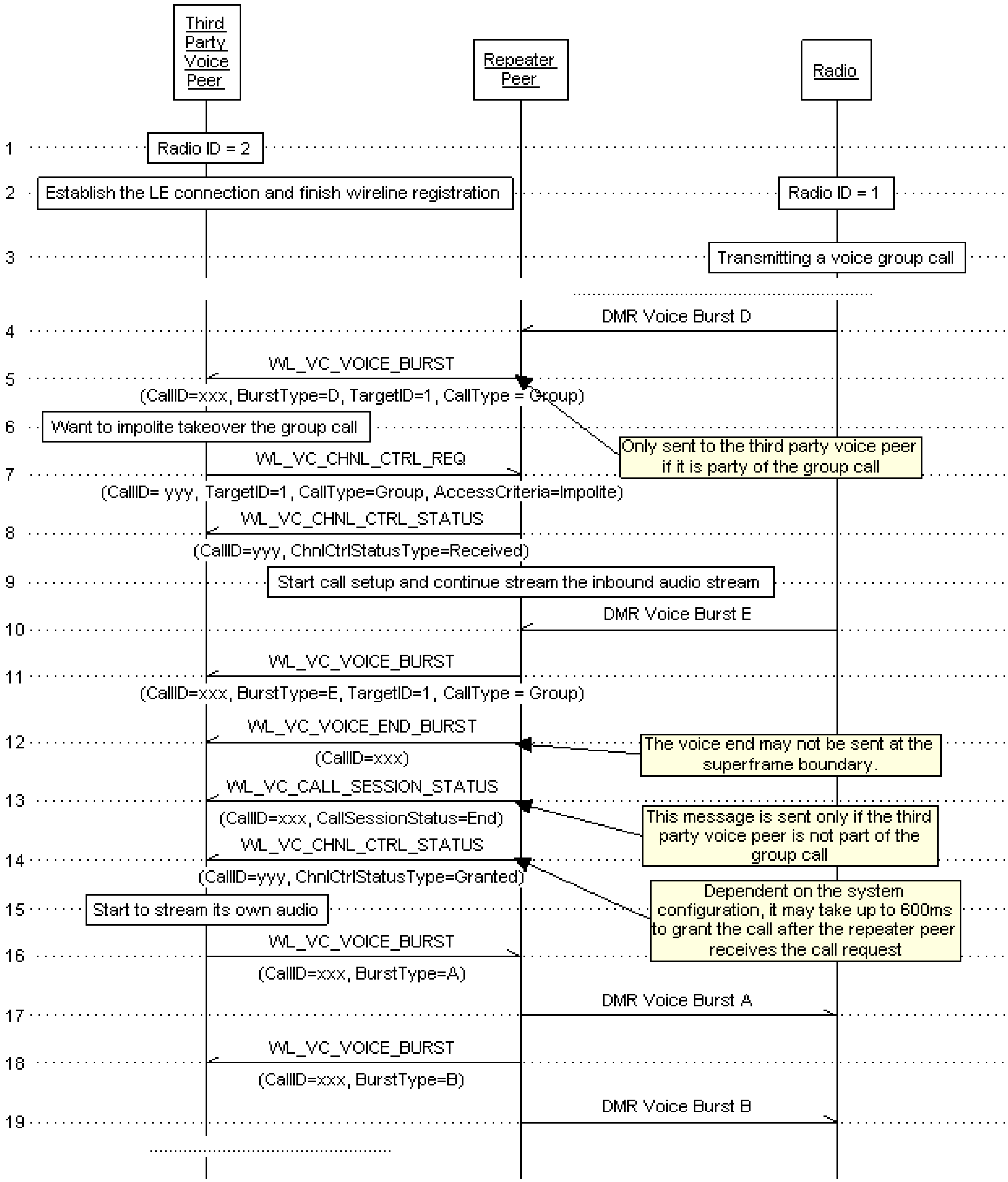


Figure 61: Application Impolitely Take Over Call

3.5.12 Transmit Interrupt

The Transmitter Interrupt feature allows a radio or a third party application to shutdown an in-progress voice transmission, and potentially initiate a new transmission. It has four unique variations to support different use cases. These unique variations are:

- Voice Interrupt: Allows a radio or third party application that is a part of a call to stop the in-progress interruptible voice transmission, and initiate its own voice transmission to the same call membership.

- Remote Radio Dekey: Allow a radio or third party application to stop an in-progress interruptible voice transmission. The radio or third party application may or may not be partied to the interrupted voice call in the conventional system. In both the Linked Capacity Plus and Capacity Plus system, the radio can stop an ongoing call only on the channel where it is present, the third party application must receive the call by either monitoring or being party to the call. Only on a “busy rest channel”, the radio can stop the on-going call even it is not party to the call.

- Emergency Voice Interrupt: Allow a radio or third party application to stop any in-progress interruptible voice transmission and initiate its own emergency transmission. In Capacity Plus and Linked Capacity Plus systems, this feature is used to stop the interruptible transmission in the following two cases:

    - If a call is active for the same talkgroup on channel ‘c’ then a radio or third party application starts the emergency call on channel ‘c’.
    - If all channels are busy, a radio or third party application starts an emergency call over the busy Rest channel.

  If the ongoing transmission is not interruptible, then the radio or third party application transmits impolitely over the ongoing call.

- Data Over Voice Interrupt (DoVI): Allows a third party application to stop any in-progress voice transmission and initiate its own data transmission. Since all the data call shall be initiated from the MNIS interface, please see reference [7] for more details on DoVI.

3.5.12.1 Remote Dekey

To remotely dekey an on-going call, the third party application sends a WL_VC_CHNL_CTRL_REQUEST to the source repeater peer directly, which is receiving the call over the air from the radio. The third party application shall not send the remote dekey request to the Site Peer. It shall find the source repeater and the slot number from the WL_VC_VOICE_BURST message when receiving the on-going audio stream. The WL_VC_CHNL_CTRL_REQUEST shall have the following setting:

- SlotNumber: the slot number the to-be-dekeyed call is on-going

- CallType: Remote Dekey
- SourceID: the Radio ID of the third party application
- TargetID: the Radio ID of the radio which is transmitting
- AccessCriteria: Polite
- Preamble Duration: 0
- CSBK Parameters: blank

Figure 62 shows an example message sequence. After successfully stop the on-going transmission, the third party application can either start a new call using the WL_VC_CHNL_CTRL_REQUEST or does nothing. If the on-going transmission is not interruptible, the repeater peer rejects the remote dekey request from the third party application.

In case a radio remotely dekeys the on-going call, the third party application is receiving the on-going call, it receives the call stream end indication of WL_VC_VOICE_END_BURST message, and the call session end indication of the WL_VC_CALL_SESSION_STATUS message with Call Ended status.

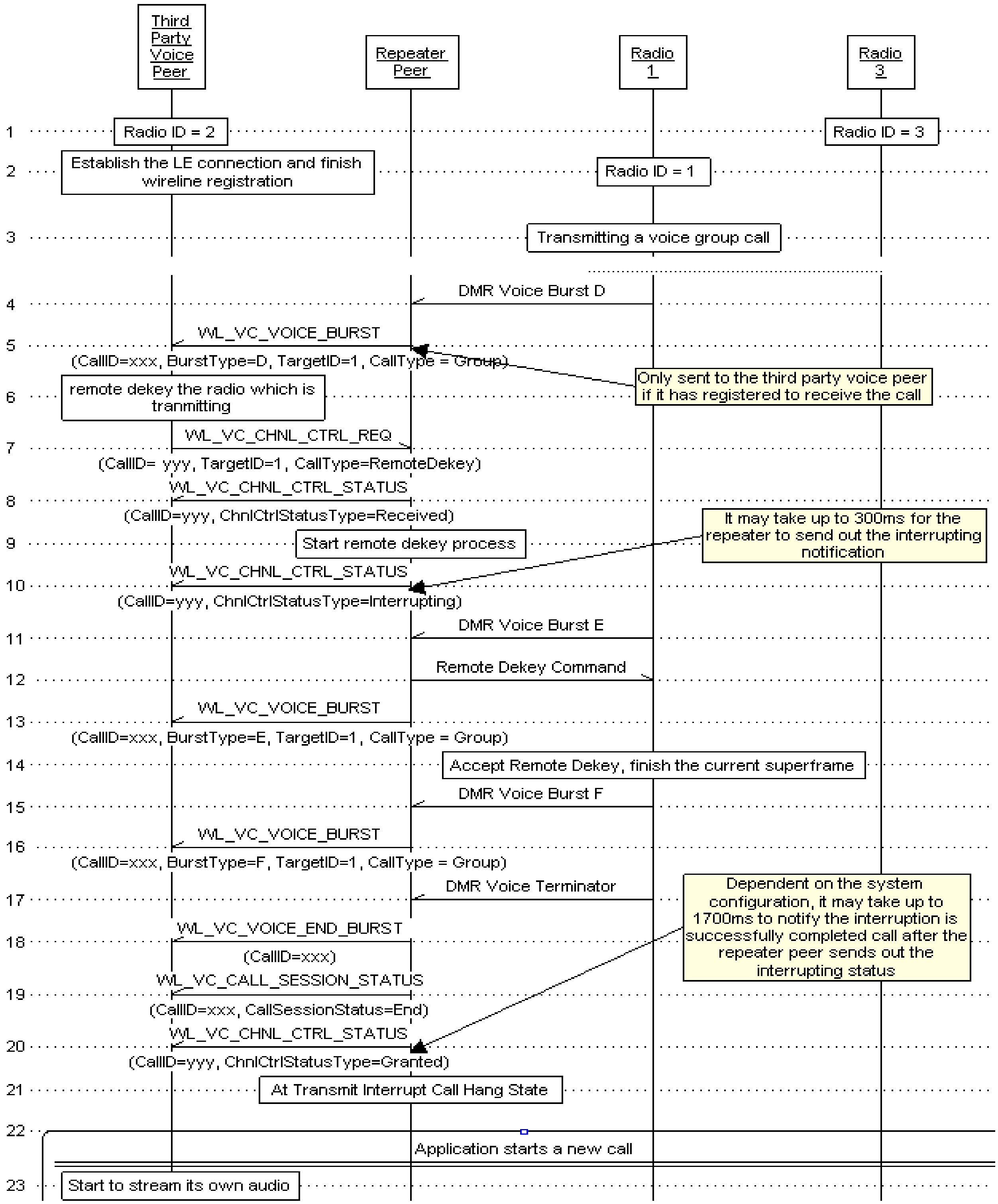


Figure 62: Application Remotely Dekey On-Going Call Voice Call

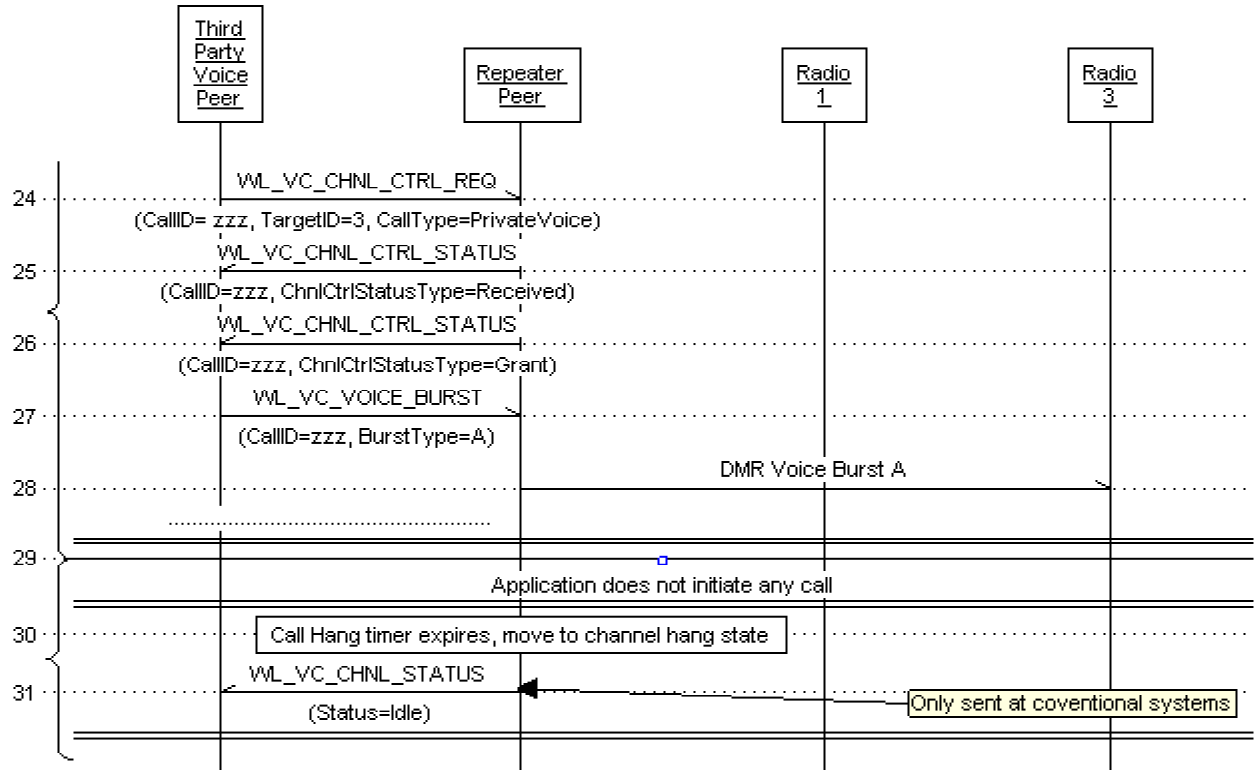


Figure 63: Application Remotely Dekey On-Going Call Voice Call (Cont)

3.5.12.2 Voice Interrupt

To interrupt an on-going call which the third party application is partied to, the third party application sends a WL_VC_CHNL_CTRL_REQUEST to the source repeater peer, which is receiving the call over the air from the radio. The WL_VC_CHNL_CTRL_REQUEST shall have with the following:

- SlotNumber: the slot number the to-be-dekeyed call is on-going
- CallType: same as the on-going call
- SourceID: the Radio ID of the third party application
- TargetID: if the on-going call is private call, set to the Radio ID of the radio which is transmitting; if group call, same as the on-going call
- AccessCriteria: Interrupt
- Preamble Duration: 2
- CSBK Parameters: blank

Figure 64 shows an example message sequence where after successfully stopping the on-going transmission, the third party application starts a new audio stream to the same call parties.

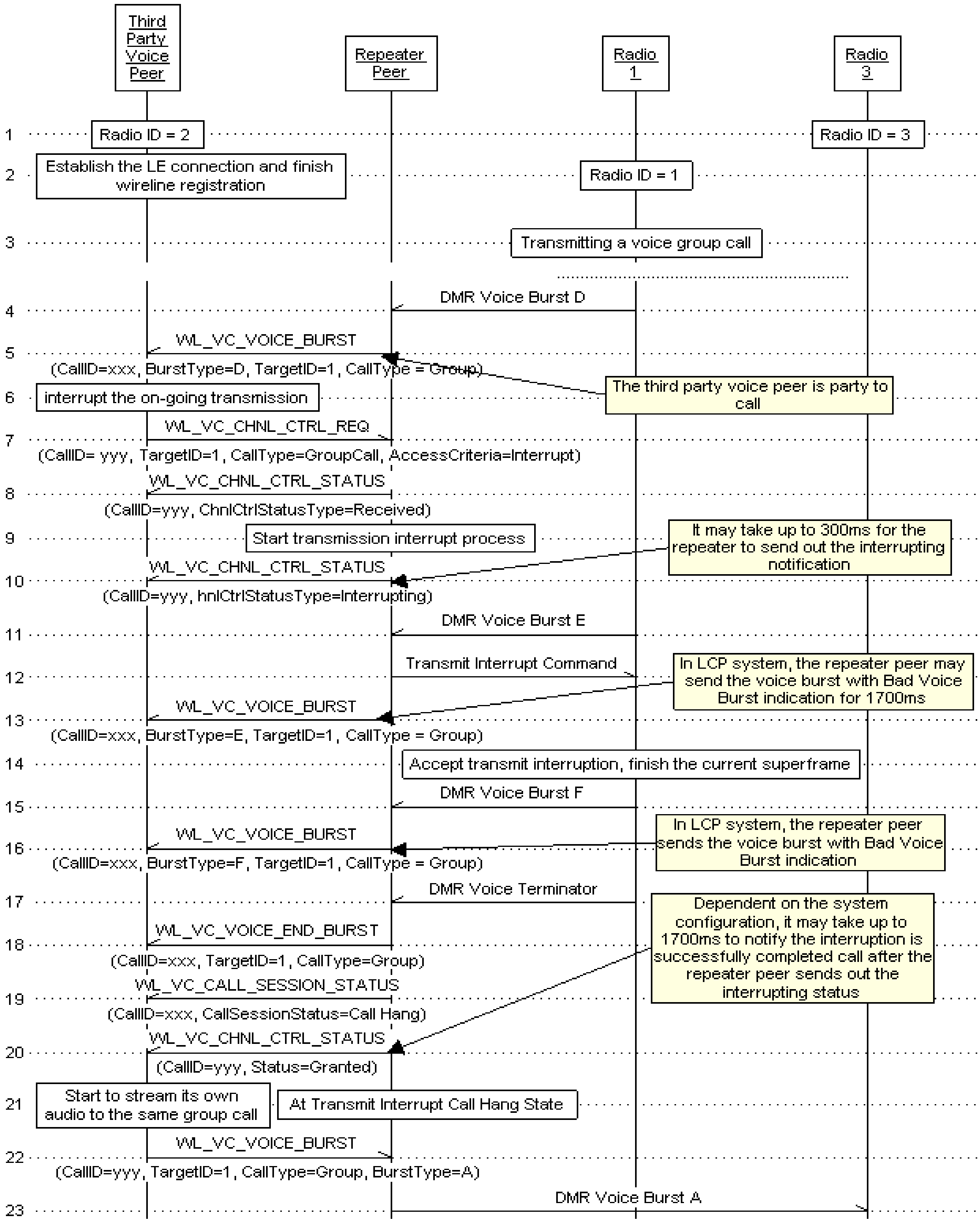


Figure 64: Application Interrupt On-Going Call Voice Call

3.5.12.3 Emergency Voice Interrupt

When the third party application wants to initiate its own emergency call while the channel is occupied, the third party application sends a WL_VC_CHNL_CTRL_REQUEST to the source repeater peer directly, which is receiving the call over the air from the radio. The third party application shall not send the request to the Site Peer. It shall find the source repeater and the slot number from the WL_VC_VOICE_BURST message when receiving the on-going audio stream. The WL_VC_CHNL_CTRL_REQUEST shall have the following setting:

- SlotNumber: the slot number the to-be-dekeyed call is on-going
- CallType: Emergency CSBK Alarm Request
- SourceID: the Radio ID of the third party application
- TargetID: the emergency talkgroup ID
- Access Criteria: Interrupt
- Preamble Duration: 2 for trunking system; in conventional system, follow the radio configuration which is normally greater than 2.
- CSBK Parameters: 8-byte Emergency CSBK structure (see section 7.1 for the structure detail)

If the on-going call is not interruptible, the repeater peer still accepts the request and places the third party application audio stream impolitely over the ongoing call.

Figure 65 shows an example message sequence when the on-going call is interrupted by the emergency call from the third party application.

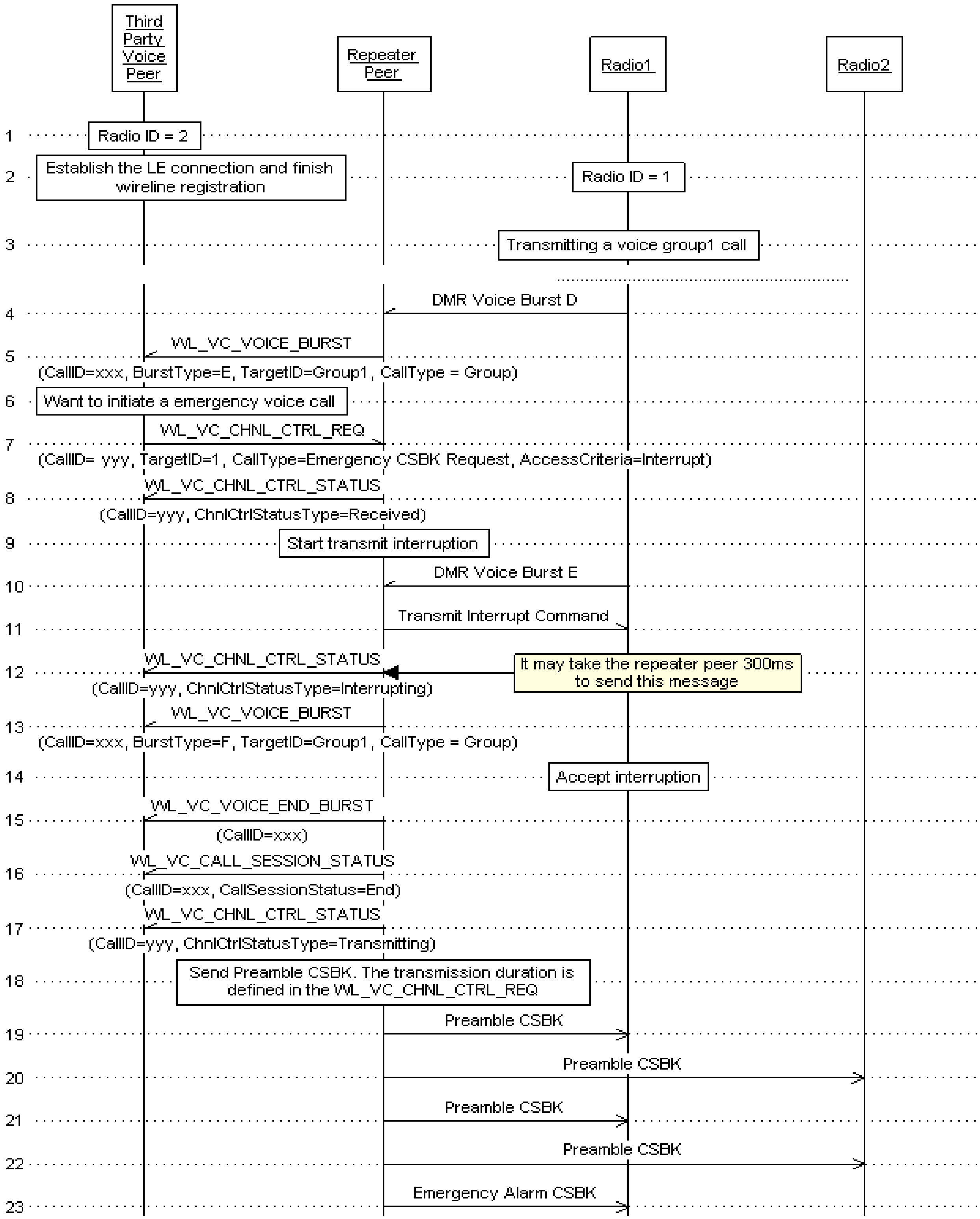


Figure 65: Application Initiate Emergency Interruption Over On-Going Call Voice Call

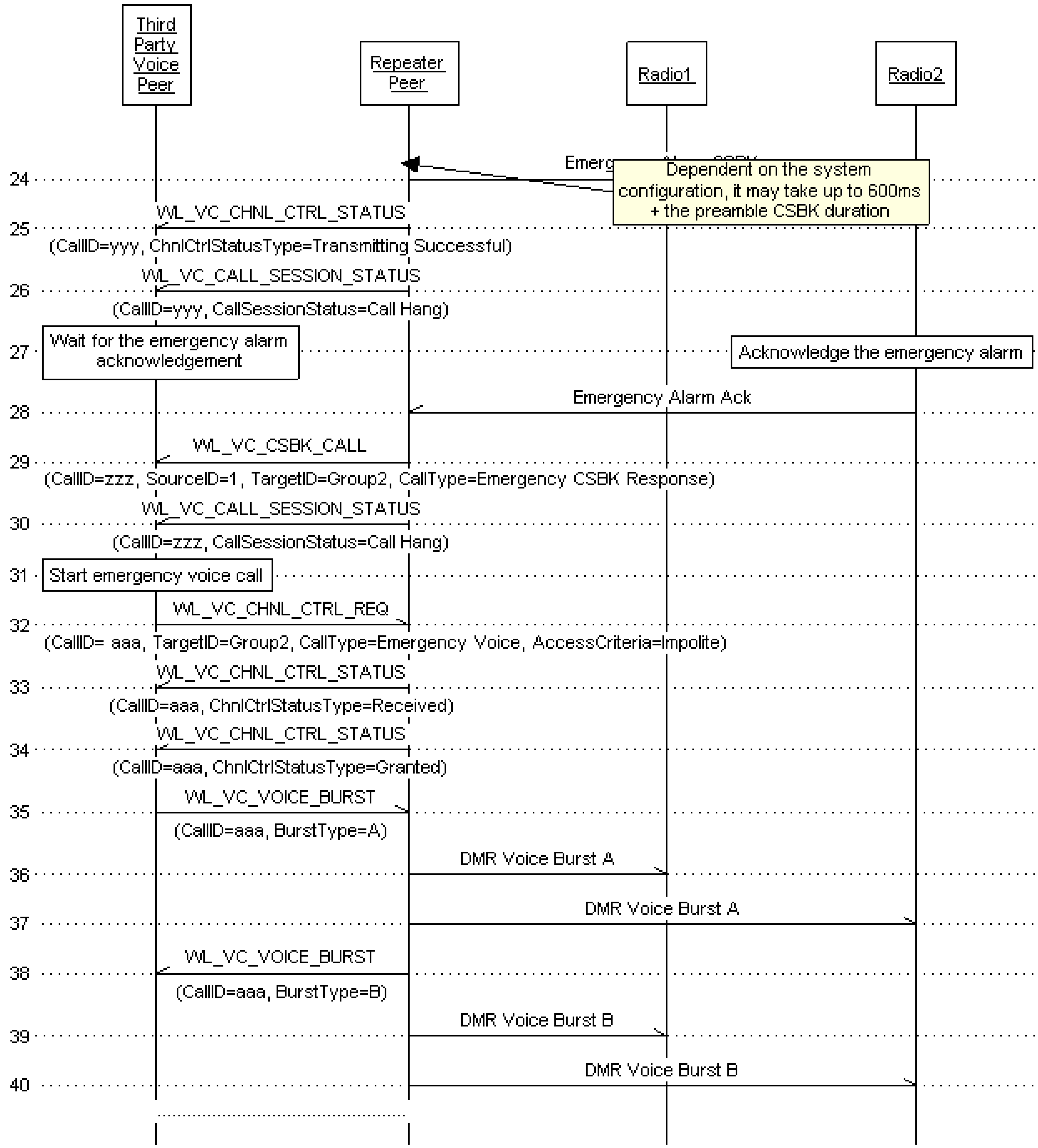


Figure 66: Application Initiate Emergency Interruption Over On-Going Call Voice Call (Cont)

In case a radio initiates the emergency voice interruption on the on-going call, the third party application is party to the on-going call, it receives the call stream end indication of WL_VC_VOICE_END_BURST message, and the call session end indication of the WL_VC_CALL_SESSION_STATUS message with Call Ended status.

3.6 *Wireline Control Calls (CSBK)*

3.6.1 Call Session

Before we introduce the detailed call setup procedure, the following key concepts deserve a clear definition and clarification:

- **Call Session:** A CSBK call starts with a CSBK request and ends with a CSBK response. One call session contains the CSBK request and the CSBK response.
- **Call Session Hang Status:** When the repeater sends the CSBK request, it starts the Call Hang Time, moves to the Call Hang state and waits for the CSBK response.
- **Call Session End Status:** When either the repeater Call Hang Time expires or the CSBK response is received, the call session is considered to be over.

In the conventional systems, after the Call Hang Time expires, the repeater peer may still accept the CSBK response from the third party application and repeat it over the air if the channel is still idle. In the trunking system, after the Call Hang Time expires, the current channel is no long allocated for the CSBK call, the repeater peer rejects the CSBK response. Since the radio will re-send the CSBK request if it does not receive CSBK response in time, it is recommended that the third party application shall not send the CSBK response after the call session ends.

3.6.2 Call Routing

Similar to the voice call, to send/receive a CSBK call to/from a radio, the third party application must finish the Wireline Registration with the repeater peers in the system. Please refer to section 3.2.1for the detailed routing rules.

3.6.3 Roles in the CSBK Call

There are different responsibities owned by the third party application and repeater peers in a CSBK call session. Table 16 summarizes the responsibilities of repeater peer and third party application when radio initiates a CSBK call to the third party application or the third party application initiates a CSBK call to the radio.

Call Originator	Repeater Peer	Third Party Application
Radio	• Notify the channel status change (Single Site or IPSC only) • Floor arbitration with other repeater peers • Send CSBK request to the application. • Repeat the CSBK call	• Retrieve the call information from the Call Control interface message • After receiving call session status of call hang, send the CSBK response with the call request to the repeater peer

Call Originator	Repeater Peer	Third Party Application
	over the air • Maintain the repeater call hang timer • Indicate the call session status	
Third Party Application	• Notify the channel status change • Update the call status to the third party application, e.g. request received, channel granted • Floor arbitration with other repeater peers • Generate DMR CSBK burst and repeat the CSBK call over the air • Maintain the repeater call hang timer • Send radio's CSBK response to the third party application • Indicate the call session status	• Check if the channel status is idle before sending the call request to the repeater peer • Re-send the call request if the call is not set up successfully, e.g. lack of response from repeater, other call wins the channel. • Receive CSBK response from the repeater peer. If a NACK is received, the third party application can decide if a retry is needed or not.

Table 23: Peer Responsibilities in CSBK Call

The third party application shall only initiate a CSBK call when the channel status is idle because the repeater peer cannot queue the call request. For trunking systems, the call request is always sent to the Site Peer, the third party application can initiate multiple calls into the system, but make sure initiate the new call request after the first one has been accepted by the repeater system.

3.6.4 DMR CSBK Bursts

The DMR CSBK message contains a 96-bit information field. The general structure of the CSBK message is shown in Figure 67. The CSBK message is transmitted in a single data burst over the air.

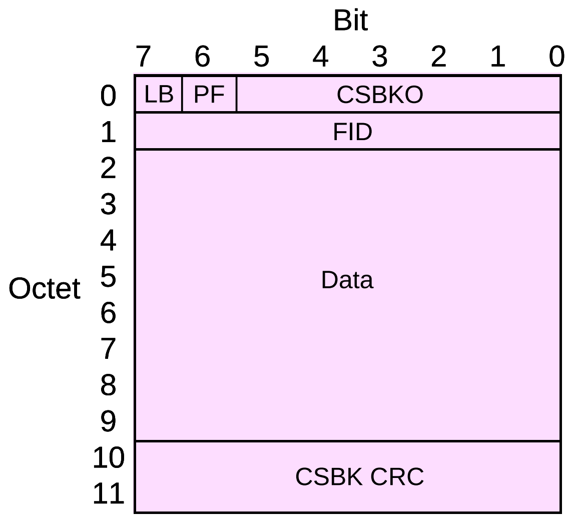


Figure 67 - CSBK message structure

3.6.5 Wireline CSBK Messages

The WL_VC_CHNL_CTRL_REQUEST is used to send either a CSBK request or a CSBK response from the third party application to a radio. The WL_VC_CSBK_CALL is used to receive either a CSBK request or a CSBK response from a radio to a third party application. Both WL_VC_CHNL_CTRL_REQUEST and WL_VC_CSBK_CALL message has a 8-byte CSBK parameter field, which has the same content as the Byte 2 – Byte 9 in Figure 67. The structure of the byte 2 – byte 9 is dependent on the type of the CSBK call. See section 7 for more details. The FID field in the WL_VC_CHNL_CTRL_REQUEST or WL_VC_CSBK_CALL message matches the FID in the Byte 1 of Figure 67. When generating the over-the-air CSBK burst, the repeater peer fills the remaining fields, e.g. CSBK CRC, in Figure 67.

The example WL_VC_CSBK_CALL with CSBK Private Call Request sent from the radio to the third party application is shown as below.

Offset	Raw Data	Description
0	0xb2	Call Control Opcode
1	0x00 00 03 e9	Application Peer ID
5	0x17	Opcode for WL_VC_CSBK_CALL
6	0x02	Slot Number where the call is received
7	0x43	Call Type: Private Call Request (unit to unit voice service request)
8	0x00 00 00 2f	Call ID for this call
12	0x00 00 01 94	Source ID
16	0x00 00 00 01	Target ID
17	0x00 00	Reserved
19	0x00	MFID

Offset	Raw Data	Description
20	0x00 00 00 00 01 00 01 94	CSBK parameter: Service Options =0x00 Reserved=0x00 Target ID: 0x00 00 01 Source ID: 0x00 01 94
28	0x01 a8	Raw RSSI value
30	0x04	Current / Accepted Wireline Protocol Version
31	0x04	Oldest Wireline Protocol Version
32	Wireline Authentication ID	Uint32
36	Wireline Authentication Signation	Uint8 * 10

Table 24: Example Message of WL_VC_CSBK_CALL with Private Call Request

The example WL_VC_CHNL_CTRL_REQUEST with Private Call Response sent from the third party application to the radio is shown as below.

Offset	Raw Data	Description
0	0xb2	Call Control Opcode
1	0x00000264	Application Peer ID
5	0x13	Opcode for WL_VC_CHNL_CTRL_REQUEST
6	0x02	Slot Number where the call is received
7	0x4c dc 12 38	Call ID for this call
11	0x44	Call type: Private Call Response (Unit to Unit Voice Service Response)
12	0x00 00 00 01	Source ID
16	0x00 00 01 94	Target ID
20	0x03	accessCriteria is impolite
21	0x00	callAttribute is clear call
22	0x00	Reserved field
23	0x 00	Preamble Duration: 0
24	0x00 00	Reserved
26	0x00 20 00 01 94 00 00 01	CSBK Parameters: Service Options =0x00 Answer Response=0x20 Target ID: 0x00 01 94 Source ID: 0x00 00 01
34	0x04	Current / Accepted Wireline Protocol Version
21	0x04	Oldest Wireline Protocol Version
22	Wireline Authentication ID	Uint32

Offset	Raw Data	Description
	Wireline Authentication Signature	Uint8 * 10

Table 25: Example Message of WL_VC_CHNL_CTRL_REQUEST with Private Call Response

3.6.6 Preamble CSBK

The DMR CSBK burst is sent in one burst over the air. To avoid missing call for the radios in sleeping mode or scanning mode, preamble CSBK bursts are sent before the CSBK burst with the interval of 60ms. The third party application can specify the duration of the preamble CSBK in the WL_VC_CHNL_CTRL_REQUEST message with CSBK request in a conventionally. Usually it is recommended to set the preamble duration to 120ms assuming only the battery saver is turned on in the target radio. If the radio has a long scanning list, the third party application shall increase the preamble duration accordingly. In the trunking system, the third party application is recommended to set the preamble duration to 120ms also in the WL_VC_CHNL_CTRL_REQUEST message.

The preamble duration shall set to 0 when the WL_VC_CHNL_CTRL_REQUEST carries the CSBK response.

3.6.7 Exception Handling

3.6.7.1 Unrecognized MFID

The MOTOTRBO repeater only supports the following two MFID values:

- Standard: 0x00
- Motorola Proprietary: 0x10

The following table shows the CSBK Opcode and its associated MFID:

CSBK Type	Value	MFID
Preamble	%111101	0x00
Negative Ack Response (NACK_RSP_U)	%100110	0x00
Unit to Unit Service Request (UU_V_REQ)	%000100	0x00
Unit to Unit Service Answer Response (UU_ANS_RSP)	%000101	0x00
Positive Ack Response (ACK_RSP_U)	%100000	0x10
Call Alert Request (CALL_ALERT_REQ)	%011111	0x10
Emergency Alarm Request (EMRG_ALRM_REQ)	%100111	0x10
Radio Monitor Command (RAD_MON_COM)	%011101	0x10
Extended Function Command (EXT_FNCT_CMD)	%100100	0x10
Extended Function Response (EXT_FNCT_RSP)	%100100	0x10

Table 26: MFID Assignment for CSBK Opcode

If a repeater peer receives a CSBK message that contains an unrecognized MFID, it will discard the message.

3.6.7.2 Lost Response

See section 3.5.9.5 for the detailed information.

3.6.7.3 Lost WL_VC_CALL_SESSION_STATUS Message

After the CSBK Request is transmitted over the air, the source repeater is expected to send the WL_VC_CALL_SESSION_ STATUS message to indicate it is in the Call Hang state now, then the third party application can send the WL_VC_CHNL_CTRL_REQUEST for the CSBK response or wait for the CSBK response from the radio. The CSBK call hang duration is defined in section 5.2.

The WL_VC_CALL_SESSION_ STATUS could be lost in the network, the third party application can assume the repeater is in the Call Hang if it does not receive WL_VC_CALL_SESSION_STATUS message after 120ms since it receives the WL_VC_CSBK_CALL message or the WL_VC_CHNL_CTRL_STATUS with status of transmission success.

3.6.7.4 Source ID for Emergency Alarm Response

The Emergency Alarm Request is a group CSBK call. In an Emergency Alarm Request from the radio, the source ID is the radio's ID and the destination ID is the emergency group ID. When the third party application sends the Emergency Alarm Response, the destination ID has to be the Radio's ID which sends the Emergency Alarm Request. Before R2.3, the source ID has to be the emergency group ID. After R2.3 it is recommended to set the source ID to be the third party application's Radio ID.

3.6.8 CSBK Call Example Message Sequence

3.6.8.1 CSBK Call in Conventional Repeater System

In a conventional system, the slots are statically assigned. In a single site conventional system, the third party application has to send WL_VC_CHNL_CTRL_REQUEST to the repeater peer and specify the slot number. In the IP Site Connect system the slot can be either wide area or local area. When the call is a local area call at certain site, the third party application has to send the WL_VC_CHNL_CTRL_REQUEST to the specific repeater peer at that site and at that slot. To initiate a wide area call, the third party application can send the WL_VC_CHNL_CTRL_REQUEST to any of the repeater peer which is the member of the wide area channel. This applies for transmitting CSBK request and CSBK response.

The repeater peer is responsible for the call setup in the system on behalf of the third party application. If the call is not successfully granted as indicated by the WL_VC_CHNL_CTRL_STATUS message, the third party application shall re-send the WL_VC_CHNL_CTRL_REQUEST with a different Call ID after the channel becomes idle.

The repeater has two slots which are independent. The third party application can simultaneously initiate two calls at different slots.

In the example below, the radio sends a call alert request to the third party application. The third party application does not send the call alert response. The radio retries and receives the response from the third party application.

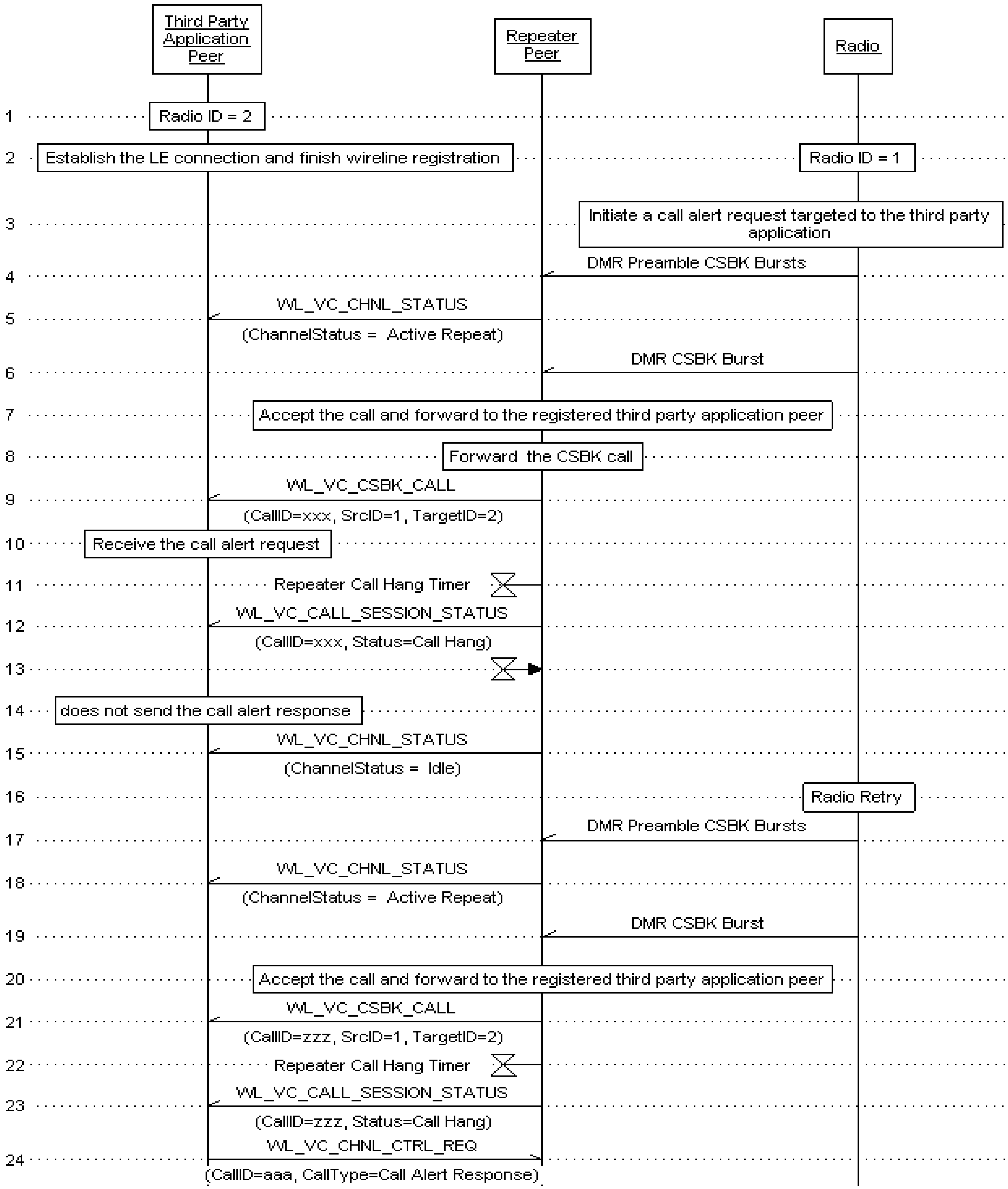


Figure 68: Receiving Call Alert at Single Site Repeater System

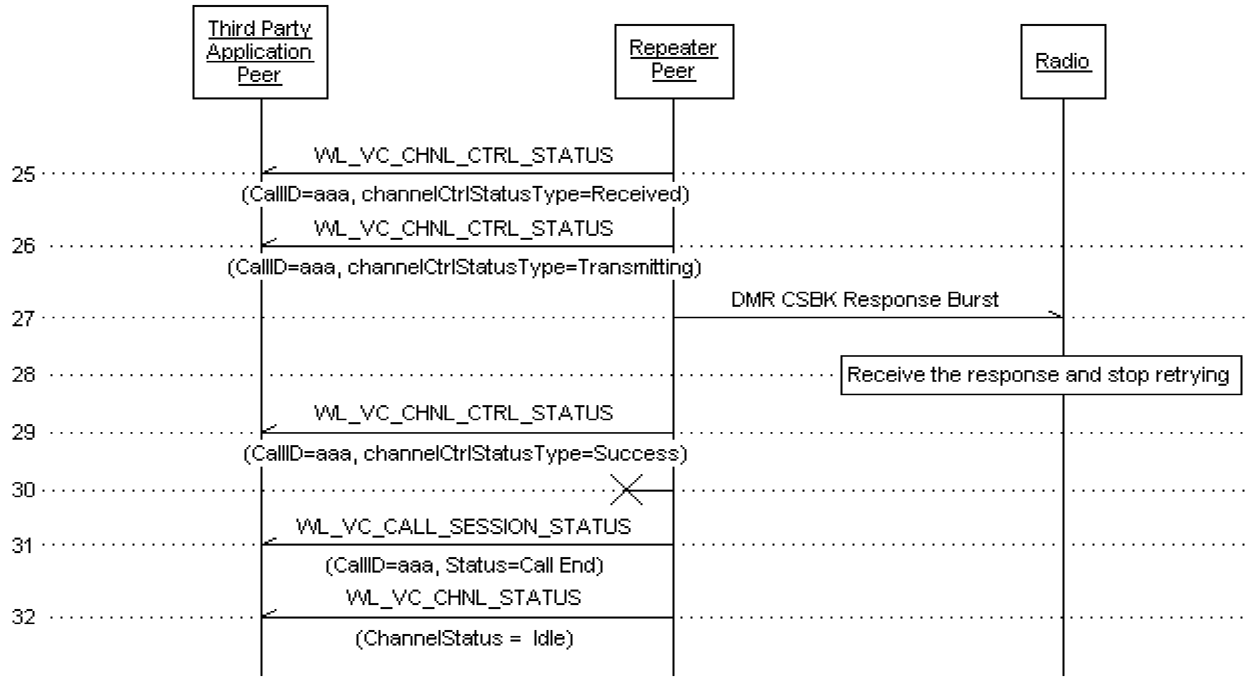


Figure 69: Receiving Voice Call at Single Site Repeater System (Continue)

3.6.8.2 CSBK Call In Linked Capacity Plus and Capacity Plus System

The call setup procedure for trunking systems is identical at the Call Control interface. Therefore in this section unless specifically pointing out, we will treat the Capacity Plus system as a Linked Capacity Plus system with a single site.

Very similar to the voice call in the trunking system, no matter it is the first try or a later retry, the third party application shall always send the WL_VC_CHNL_CTRL_REQUEST with the CSBK request to the Site Peer in the Linked Capacity Plus system, or in the Capacity Plus system. The third party application does not need to specify the slot number in the WL_VC_CHNL_CTRL_REQUEST with the CSBK request. The repeater peer acting as the Site Peer handles the call setup on behalf of the third party application. Once the call is granted, the repeater peer selects the new Rest Channel (it could be the other slot in this repeater), and notifies the third party application the call session status for this call with the WL_VC_CALL_SESSION_STATUS. The third party application can start to wait for the response from the target radio.

When the third party application receives the message WL_VC_CSBK_CALL with a CSBK request from a radio, the source IP address and source UDP port in the WL_VC_CSBK_CALL message tells which repeater is allocated for the call. This is the WAN IP address/UDP port of the repeater peer. The SlotNumber in the WL_VC_CSBK_CALL tells which slot of the repeater is allocated for the call.

The third party application shall use the allocated repeater peer's WAN IP address/UDP port as the target and the allocated slot in the WL_VC_CHNL_CTRL_REQUESTUEST. The Site Peer IP address/UDP port shall only be used when sending CSBK request from the third party application to a radio.

Similar to a private voice call, when sending a CSBK request to a radio, the third party application can send the WL_VC_CHNL_CTRL_REQUEST directly to the Site Peer where the radio currently locates and specify "Do not Forward to Remote Sites" in the CallAttribute field so that the call is set up locally without involving the remote sites.

If for some reason, the third party application does not know where the radio is, it can still send the WL_VC_CHNL_CTRL_REQUEST to anyone of the Site Peers.

Even though the Linked Capacity Plus and Capacity Plus system supports multiple calls at the same time, the third party application shall not initiate the second call until it receives the WL_CHNL_CTRL_STATUS with Transmitting status and waits for another 120ms. The repeater peer sending the WL_CHNL_CTRL_STATUS needs the 120ms to finish the rest channel movement. If the third party application sends the WL_VC_CHNL_CTRL_REQUEST before the rest channel movement finishes, the repeater peer will sends a WL_VC_CHNL_CTRL_STATUS with failure code of NON_REST_CHANNEL_REPEATER.

For channel access restriction in the trucking system, see Table 19 for more details.

In the example below, the third party application sends a remote monitor request to a radio. The radio accepts the request and keys up for the duration defined in the remote monitor request.

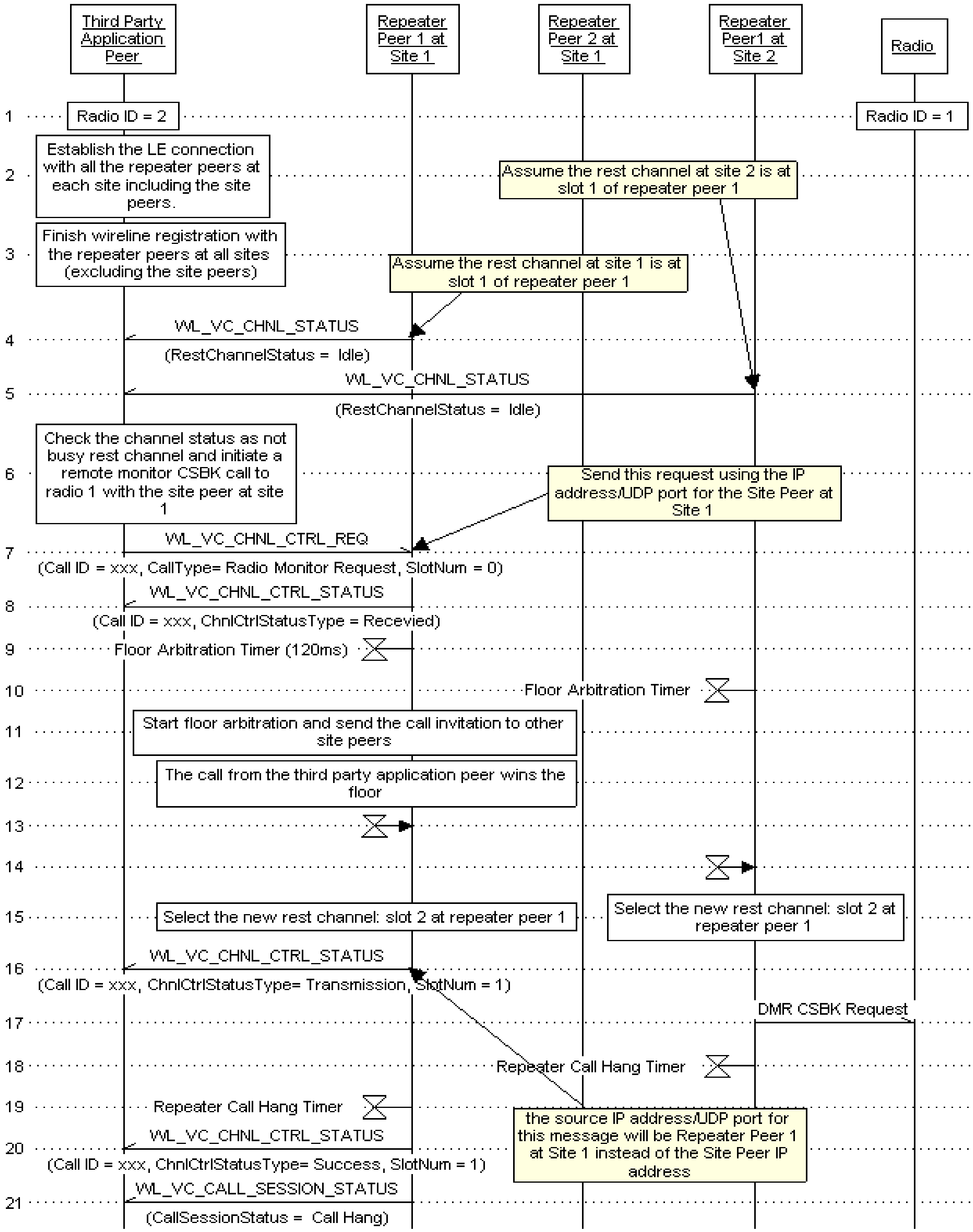


Figure 70: Application Initiate Remote Monitor Command to Radio in LCP System

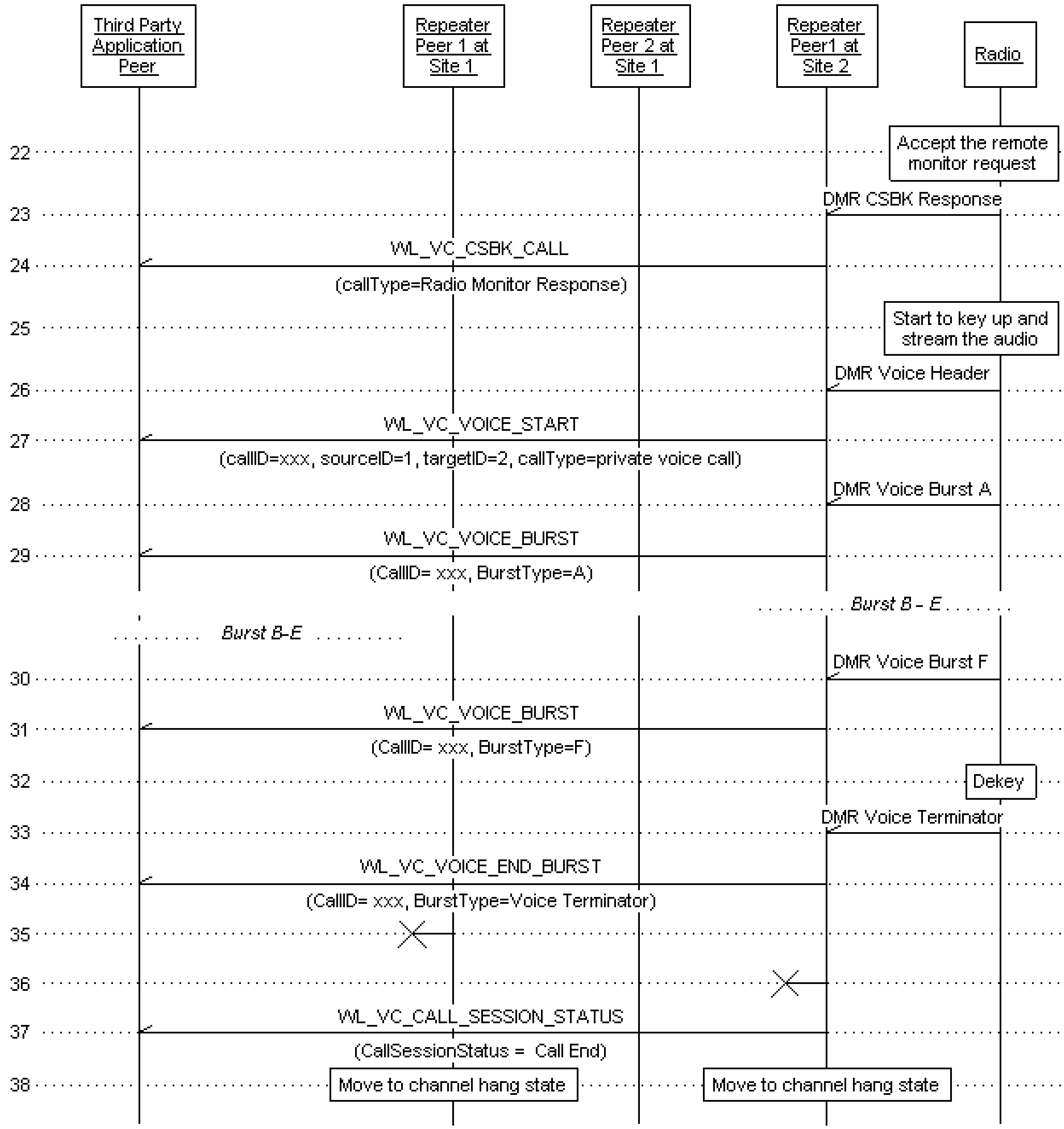


Figure 71: Application Initiate Remote Monitor Command to Radio in LCP System (Cont)

3.6.9 IP Console Inhibit (Application to Radio)

The third party application uses the IP Console Inhibit / Un-inhibit commands to disable / enable subscriber at a conventional channel or at the Linked Capacity Plus/Capacity Plus channel.

When the radio receives the IP Console Inhibit command:

- In the Capacity Plus system or Linked Capacity Plus system, the radio cannot access any of the trunked channels in the system.
- In IP Site Connect WAC, the radio cannot access the WAC channel no matter which site it goes. It can access other WAC or LAC when the user changes the channel.
- In IP Site Connect LAC, the radio cannot access the LAC any more. It can access other channel when the user changes the channel.
- In Single Site system, the radio cannot access the channel where it received the IP Console Inhibit command. It can access other channel when the user changes the channel.

The third party application can maintain an unauthorized radio list. Whenever the third party application detects the unauthorized radio access, it can use the IP Console Inhibit command to disable the radio. The IP Console Inhibit / Un-inhibit commands are introduced in R1.7 to overcome the limitation of Radio Inhibit command: the radio can ignore the Radio Inhibit command if the Radio Disable Decode is not selected when programming the radio.

Considering Radios with firmware before R1.7 does not support the IP Console Inhibit command, the XCMP Repeater Disable command is recommended to temporarily disable the over-the-air repeating function of the repeater slot which is accessed by the unauthorized subscriber. The XCMP repeater enable command can be sent to resume the repeater over-the-air transmission function so that other radios can use the channel resource. Without receiving the XCMP Repeater Enable command, the repeater remains in the disabled state for a maximum duration of 30 seconds. For details on how to use the XCMP Repeater Disable / Enable commands, refer to Section 5.6 of Reference [3]. In summary, the third party application shall use the repeater call monitoring service or call streaming service to detect the unauthorized access, use the IP Console Inhibit / Un-inhibit command to disable the radio and use the XCMP Repeater Disable / Enable command to disable the repeater slot temporarily. For details on the call monitoring service, see Reference [5] for more details.

When the radio receives the IP Console Inhibit command, it displays”channel denied” on its screen, and cannot transmit any call or receive any call except the IP Console Un-inhibit command. When the radio moves to a different channel, it becomes active. Once back to the inhibited channel, the radio becomes inhibited. The radio remains inhibited until it receives an IP Console Un-inhibited command over the air. Power cycle cannot move the radio out of the inhibited state.

When a radio is in the inhibited channel, scan can be started. But the radio cannot land on the inhibited channel for any activities except receiving IP Console Inhibit / Un-inhibit commands to send the acknowledgement.

When a radio is disabled by the Radio Disable command, it has no ergonomic display or tones. Changing to a different channel or power cycle cannot move the radio out of the disabled state. The radio cannot transmit or receive any call except the Radio Enable command.

4 Voice Privacy

On a MOTOTRBO digital channel, there are two types of privacy that can be configured in the MOTOTRBO subscriber: Basic Privacy and Enhanced Privacy. Both Basic Privacy and Enhanced Privacy are a software-based scrambling solution to prevent eavesdropping on the voice or data payload traffic. The receiving subscriber must have the same Basic Privacy Key for the Basic Privacy or the same Key Value and Key ID for the Enhanced Privacy as the transmitting subscriber to unscramble the privacy-enabled voice call or to receive the privacy-enabled data transmission.

Both Basic Privacy and Enhanced Privacy protect the information by not allowing the voice to be heard or the data to be read except by the intended receivers. The privacy methods do not provide any mechanism to authenticate the receivers or protect the integrity of the information.

Similar to the control station, the MNIS conducts the privacy processing on behalf of the third party data application. Please refer to the MOTOTRBO Network Interface Service Development Guide ADK for the further information on MNIS and the privacy configuration. The following two sections provide detailed descriptions on how the privacy information is stored within the Call Control Interface message so that a third party application can correctly unscramble the privacy enabled voice traffic.

4.1 Basic Privacy

4.1.1 Privacy Processing

In Basic Privacy, both the Sender and the Receiver have the same pre-configured Key. They generate a key-stream by repeatedly concatenating the Key to the length of the information. For example, when the sending information is 6 bytes, a 4 byte key (0x12345678) is concatenated to a 6 byte key stream (0x123456781234) to match the length of the sending information.

Figure 72 illustrates the information processing for both the Sender and the Receiver. The sender eXclusive-ORs (XOR) the plain information with the key-stream to generate the protected information and sends it over the air. The Receiver XORs the protected information with its key-stream to generate plain information.

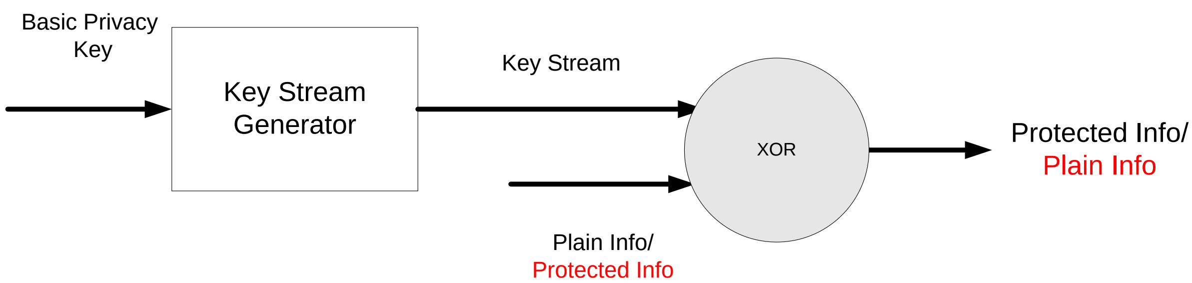


Figure 72: Basic Privacy Protection Processing

The Exclusive OR operation is done by aligning the most significant bit of both the payload and the key stream, where they are in big-endian order. Figure 73 shows the bit order of the 49-bit voice frame. In the case of voice privacy processing, the bit 40 to bit 37 in the voice frame is not exclusive-ORed with the corresponding bits of the key string in both scrambling and de-scrambling processes. Each 49-bit AMBE encoded voice frame shall be processed independently. In other words, a new key stream is generated for each voice frame.

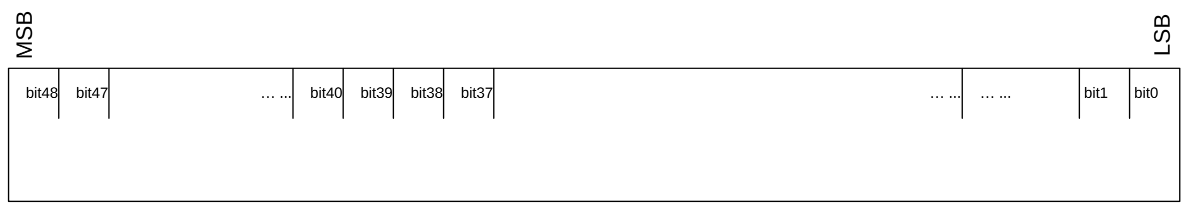


Figure 73: Voice Frame Bit Order

Since the same key is pre-configured in both the Sender and the Receiver, and the entire protection processing only depends on the key-stream, there is no transportation of key and algorithm ID in the traffic. The voice packets at the Call Control interface indicate if the voice call is protected by Basic Privacy. The repeater peer updates the Link Control in the voice header or embedded LC to indicate Basic Privacy. The third party application can use different keys on the voice calls targeted to different radios. But it has to maintain the key assignment internally to reflect what is stored in the radios.

Table 27 lists the Basic Privacy processing result based on the configuration in the transmitting radio and the receiving radio. In Table 27, Basic Privacy Key Index 1 means the Privacy Type is set to Basic, and the Basic Privacy Key is set to 1 on the CPS Privacy screen, and the Privacy field is enabled on the CPS Channel screen. Enhanced Privacy Key 1 means the Privacy Type is set to Enhanced on the CPS Privacy screen, the Privacy field is enabled and the Privacy Alias is set to Privacy Key 1 on the CPS Channel screen. Yes means the receiving radio can successfully decrypt the call; No means the receiving radio cannot decrypt the call.

Rx Radio / Tx Radio	Without Privacy	Basic Privacy Key Index 1	Basic Privacy Key Index 1 and Turned off by Programming Key	Basic Privacy Key Index 2	Basic Privacy Key Index 2 and Turned off by Programming Key	Enhanced Privacy Key 1
Without Privacy	Yes	Yes	Yes	Yes	Yes	Yes
Basic Privacy Key Index 1	No	Yes	Yes	No	No	No
Basic Privacy Key Index 1 and turned off by programming key	Yes	Yes	Yes	Yes	Yes	Yes
Basic Privacy Key Index 2	No	No	No	Yes	Yes	No
Basic Privacy Key Index 2 and turned off by programming key	Yes	Yes	Yes	Yes	Yes	Yes
Enhanced Privacy Key 1	No	No	No	No	No	Yes

Table 27: Basic Privacy Processing Based on Configuration

When Basic Privacy is enabled for the subscribers, the third party application must provision the same key as all the subscribers. In the MOTOTRBO CPS, the encryption key value is not visible to the end user, and only the Basic Privacy Key index is configurable. The application developer has to contact the regional manager to get the Basic Privacy Key list. The Basic Privacy Key is Motorola confidential information, the third party application shall protect the key information as much as possible, e.g.the third party application user interface only displays the key index instead of the key value.

4.1.2 Voice Packet for Basic Privacy

In Basic Privacy protected voice communication, the CallAttribute field in the WL_VC_CHNL_CTRL_REQUEST, WL_VC_VOICE_START and WL_VC_VOICE_BURST packets indicates if the voice call is protected by Basic Privacy.

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RESERVED	RESERVED	noForwardingToRemoteSites	RESERVED	Privacy Type		Privacy	Interruptible Voice Call

Table 28: CallAttribute Field

The following bits in the CallAttribute field are set differently compared with the un-protected voice communication:

- Privacy = 1
- Privacy Type = 1

Please note the WL_VC_PRIVACY_BURST message is not needed for the Basic Privacy call.

4.2 Enhanced Privacy

4.2.1 Privacy Processing

In Basic Privacy, the same pre-configured key index is the only required information for the Sender and the Receiver. In Enhanced Privacy, the type of algorithm, key, and Initialization Vector (IV) have to be communicated between the Sender and the Receiver. They are sent as an additional header in the voice and data calls. To support "late entry", they are also sent in every super frame of a voice call.

A radio must have privacy enabled on the channel to transmit a privacy-enabled transmission, but this is not necessary for receiving radio(s) as long as the correct key is programmed in the radio. Privacy-enabled channels are still able to receive clear (unscrambled) transmissions.

Table 29 lists the Enhanced Privacy processing result based on the configuration in the transmitting radio and the receiving radio. In Table 29, Enhanced Privacy Key 1 (1, 2, 3) means the Privacy Type is set to Enhanced, Key ID 1, Key ID 2 and Key ID 3 are configured on the CPS Privacy screen, the Privacy field is enabled and the Privacy Alias is set to Privacy Key 1 on the CPS Channel screen. Assume the key values with the same key ID are the same in the radios. Yes means the receiving radio can successfully decrypt the call; No means the receiving radio cannot decrypt the call.

Rx Radio / Tx Radio	Without Privacy	Enhanced Privacy Key 1 (1, 2, 3)	Enhanced Privacy Key 2 (1, 2, 3)	Enhanced Privacy Key 2 (2, 3)
Without Privacy	Yes	Yes	Yes	Yes
Enhanced Privacy Key 1 (1, 2, 3)	No	Yes	Yes	No

Enhanced Privacy Key 2 (2, 3)	No	Yes	Yes	Yes

Table 29: Enhanced Privacy Processing Based on Configuration

The key selected for a channel is used to encrypt the call initiated by the radio. When the radio receives an encrypted call, it searches the configured enhanced privacy key list to find the matched key ID for the decryption.

The Enhanced Privacy only protects the voice payload. The Voice Header, Motorola Proprietary PI Header, Terminator, idle packet, Link Control, and CSBK are not protected by the Enhanced Privacy.

The radio can store up to 16 40-bit keys through the CPS. A key can be associated with one or more digital channels for transmission. The Enhanced Privacy uses ARC4 as the algorithm to generate a key-stream. To produce different encryption signatures even for the same message at different times, the Enhanced Privacy uses the IV in the key-stream generation. The IV is a random number initialized during power-up.

The strength of protection depends upon the key-stream. To make the key-stream different and unrelated for every message, the IV is updated before generating a key stream for a superframe by applying Linear Feedback Shift Register (LFSR) on previous IV.

The 40-bit key is concatenated with the 32-bit IV and is used to initialize the S-box of ARC4 by using the Key-Scheduling Algorithm (KSA) of ARC4. Both the Sender and the Receiver generate a key-stream byte by byte by applying PseudoRandom Generation Algorithm (PRGA) of ARC4 over the S-box.

Figure 74 illustrates the information processing for both the sending and receiving enhanced privacy voice call.

At the sending direction (from the third party application to radios):

- The third party application eXclusive-ORs (XOR) the plain information with the key-stream to generate the protected information.
- The third party application inserts the IV into the protected information (see section 4.2.3.1 for more details), sends the protected information along with the Enhanced Privacy key ID, algorithm ID, and the IV to the repeater peers using the Call Control interface messages
- The repeater peer updates the voice header, and generates the PI header before sending it over the air.

For the receiving direction (from radios to the third party application)

- The repeater peer extracts the Enhanced Privacy key ID, algorithm ID and the IV from the received DMR bursts, and puts them in the Call Control interface message along with the protected information.

- The third party application XORs the protected information with its key-stream to generate plain information.

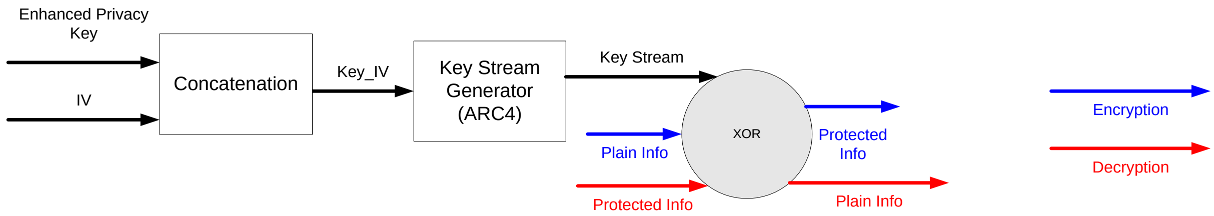


Figure 74: Enhanced Privacy Processing Flow

The Exclusive OR operation is done by aligning the most significant bit of both the payload and the key stream, where they are in big-endian order.

In the voice transmission, the length of the key stream shall be equal to the payload of a superframe (18 frames, 882 bits). Figure 73 shows the bit order of the 49-bit voice frame. Figure 75 illustrates the encryption/decryption processing on the voice frames in a voice call. Note that the IV in the WL_VC_PRIVACY_BURST is used for the key generation for the first voice super frame, the IV in the WL_VC_VOICE_BURST message with Burst F is used for the key generation for the next voice super frame. If the third party application did not receive the WL_VC_PRIVACY_BURST, it shall obtain the crypto parameters (IV, key Id, Alg Id) from the WL_VC_VOICE_BURST with Burst F, and start to decrypt on the next superframe.

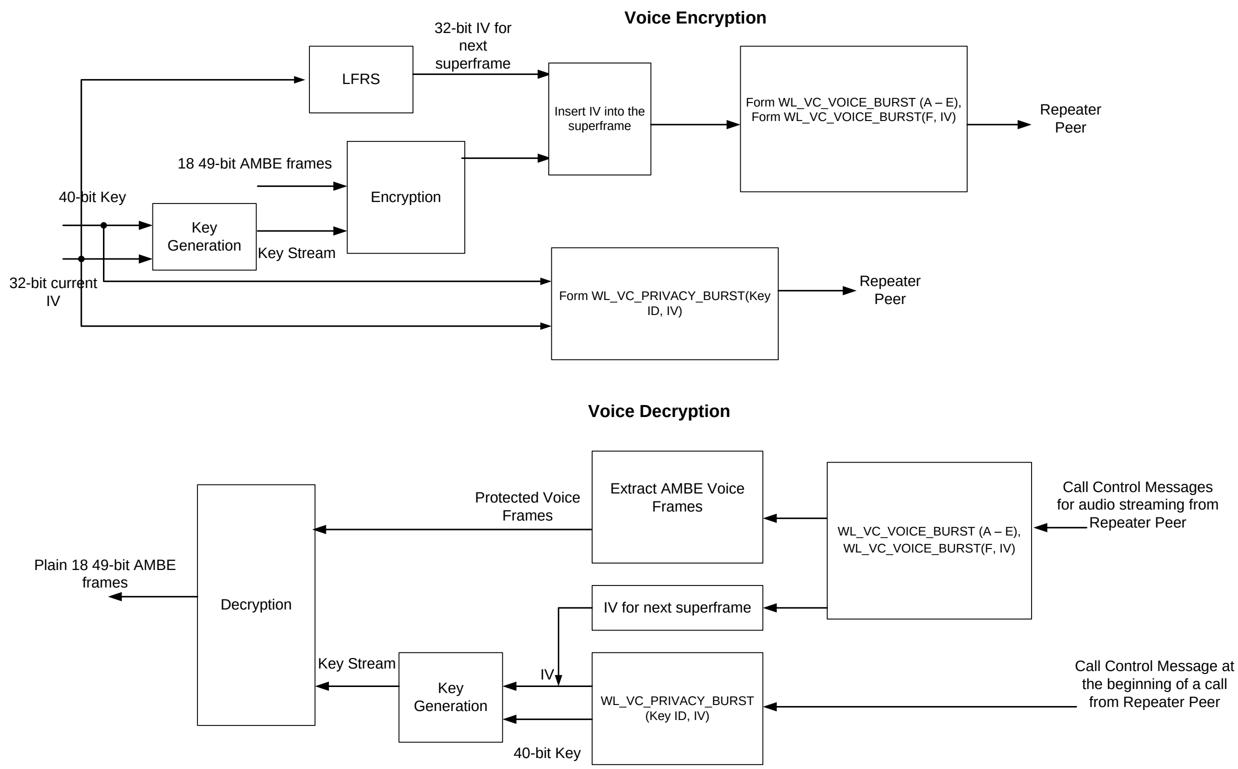


Figure 75: Encryption/Decryption Process in Voice Call

4.2.2 Privacy Algorithms

ARC4 (Alleged RC4) is used in the Enhanced Privacy key stream generation. It is the same as RC4. However, the name "RC4" is trademarked by RSA Security. The current status is that "unofficial" implementations are legal, but cannot use the RC4 name. See Reference [8] for detailed information.

ARC4 Key-Scheduling Algorithm (KSA) is used to initialize the S-box of ARC4. First, an array "S" is initialized to the identity permutation. S is then processed for 256 iterations in a similar way to the main PRGA algorithm, but also mixes in bytes of the key at the same time. The following is the key-scheduling algorithm pseudo code:

key_IV = key || IV; key_IV_length = 9; j := 0;

for i from 0 to 255 S[i] := i endfor
for i from 0 to 255
j := (j + S[i] + key_IV[i mod key_IV_length]) mod 256;
swap(S[i],S[j])
endfor

Pseudo-Random Generation Algorithm (PRGA) is used to generate the key stream. For each byte needed, the PRGA modifies the state and outputs a byte of the keystream. In each iteration, the PRGA increments *i*, adds the value of S pointed to by *i* to *j*, exchanges the values of S[*i*] and S[*j*], and then outputs the value of S at the location S[i] + S[j] (modulo 256). Each value of S is swapped at least once in every 256 iterations.

The MOTOTRBO radio discards the first 256 bytes of Keystream generated by the PRGA. We call it "drop_256" step. The third party application has to follow the same procedure to correctly encrypt/decrypt the call. The following is the pseudo code of the MOTOTRBO "drop_256" step and the PRGA:

```
// MOTOTRBO drop_256 step to discard the first 256 bytes of the Keystream
i := 0; j := 0;
for k from 0 to 255
i := (i + 1) mod 256;
j := (j + S[i]) mod 256;
swap(S[i],S[j]);
output S[(S[i] + S[j]) mod 256];
endfor

// Keystream generation for encryption/decryption

while GeneratingKeystream:
i := (i + 1) mod 256;
j := (j + S[i]) mod 256;
swap(S[i],S[j]);
output S[(S[i] + S[j]) mod 256];
endwhile
```

LFSR is used to update the IV**.** It shall be implemented in Galois configuration, where the bits that are not taps are shifted as normal. On the other hand, the taps are XOR'd with the new output, which also becomes the new input. The characteristic polynomial is "$X^{32} + X^4 + X^2 + 1$". The following is the LFSR pseudo code:

```
if (IV & 0x00000001)

        then IV = ((IV XOR 0x8000000B)>>1)| 0x80000000 /*taps at 32,4,2,1*/
else IV = IV >> 1;
```

See Appendix A for the LFSR sample implementation.

4.2.3 Message Settings for Enhanced Privacy

For a successful decryption, a receiver must know the crypto parameters (i.e. Algorithm Id, Key Id, and IV) used by the sender. Over the air, a PI header is added after the voice LC header, which contains the crypto parameters. The repeater peer uses the crypto information in the WL_VC_PRIVACY_BURST to generate the PI header over-the-air.
To facilitate "late entry" in voice call, the crypto parameters are also sent in every super frame. The WL_VC_VOICE_BURST with Burst F contains IV, Key Id and Algorithm Id.

In Enhanced Privacy protected voice communication, the following fields in the Call Control interface messages are set differently compared with the un-protected voice communication:

- The MFID is set to “Motorola Proprietary Feature” (00010000_2) in the WL_VC_VOICE_BURST and WL_VC_PRIVACY_BURST message.
- The “Privacy” bit of “Call Attribute” field is set to 1 to indicate it is privacy call in WL_VC_CHNL_CTRL_REQUEST and WL_VC_VOICE_BURST message
- The “Privacy Type” bit of “Call Attribute field is set to (10_2) to indicate it is enhanced privacy in WL_VC_CHNL_CTRL_REQUEST and WL_VC_VOICE_BURST message
- The “Privacy” bit of “Service Options” is set to 1 in the WL_VC_VOICE_BURST message (for receiving call only). The third party application does not need to set this field when initiating a call.
- The IV, Key Id and Algorithm Id in the WL_VC_PRIVACY_BURST, and WL_VC_VOICE_BURST with Burst F shall be filled. The Algorithm ID shall be set to 1 (ARC4). The IV is for the next super frame, which is the LFSR result of the current IV.

Steps to decode Enhanced Privacy voice:

1. Sbox Initializations using the input key and the IV
   1.1 Standard RC4 method for initializing the Sbox:
   1.2 MOTOTRBO "drop_256" step

2. Within one superframe, there are 18 voice frames. Each voice frame has 49-bit AMBE voice data. For each voice frame,
   2.1 As shown in the diagram below, pack the 49-bit voice data to 7 bytes character array. In the last byte fill the bit 6-bit 0 to 0. The order to decode the audio data packed in bytes shall be from byte[0] to byte[6] when XORing with the key stream.
   2.2 Use the standard Pseudo-Random Generation Algorithm to decrypt the 7-byte data.
   2.3 Extract the 49-bit AMBE data from the 7-byte decrypted data.
   2.4 Repeat the 2.1 – 2.3 for each voice frame.

3. After finishing all the voice frames in one superframe, use the new IV and the input key to initialize the Sbox as in Step 1) and prepare for the next superframe decoding.

	msb							lsb
Byte[0]	d48	d47	d46	d45	d44	d43	d42	d41
Byte[1]	d40	d39	d38	d37	d36	d35	d34	d33
Byte[2]	d32	d31	d30	d29	d28	d27	d26	d25
Byte[3]	d24	d23	d22	d21	d20	d19	d18	d17
Byte[4]	d16	d15	d14	d13	d12	d11	d10	d9
Byte[5]	d8	d7	d6	d5	d4	d3	d2	d1
Byte[6]	d0	0	0	0	0	0	0	0

4.2.3.1 Bit-stealing and IV insertion in a voice superframe

The 32 bit IV after applying CRC and FEC is sent by stealing 4 bits from each AMBE+2 vocoder frames. The 4 bit CRC is calculated using the polynomial x4 + x + 1. During the 4-bit CRC calculation, set the register's initial value to 0, reverse the 32-bit IV before multiplying with the polynomial, and XORed the final register value with 0x0F. For example, the 4-bit CRC for IV of (24, 71, DF, B6) is 0101b by first being reverted to 10110110110111110111000100100100b (B6, DF, 71, 24), then multiplied with the polynomial and XORed with 0x0F,

The IV is reversed and concatenated with the 4-bit CRC to form the 36 bits, which are splited into three groups of 12 bits and a (24, 12, 8) Golay code is applied as FEC. Please note all the IVs in Figure 76 is reverted (the least significant byte becomes the most significant byte). See Appendix B for Golay(24, 12, 8) matrix definition.

Interleaving is used to spread each of the Golay codes over the entire voice superframe for resistance to fades. At the destination, all three codes have to be correctly decoded for IV to be useful.

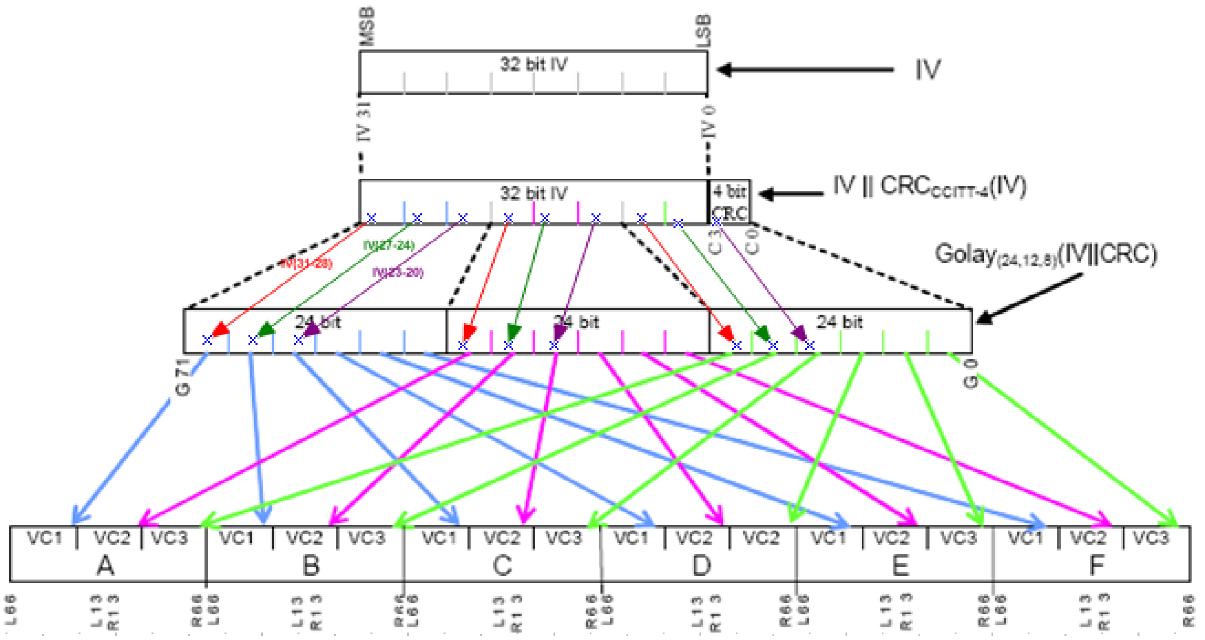


Figure 76: KeyID/AlgID Embedded LC

36-bit IV+CRC is Golay encoded to obtain 72-bits of FEC encoded data. As shown in the figure above, the FEC encoded IV bits are named as G71 (MSB) through G0 (LSB). These 72 bits are then inserted in one voice superframe by stealing 4 bits in every 20-ms AMBE+2 encoded voice frame.

The following is an example of Golay transformation:

IV (00, 3E, F3, 4C), its CRC4^0xF is 1111b

000000000011b (003) results in Golay code: 000111010101b (1D5)
111011110011b(EF3) results in Golay code: 111100011010b (F1A)
010011001111b(4CF) results in Golay code: 111011111100b (EFC)
Therefore, the 72 Golay encoded bits from G71 to G0 bit are:
00, 31, D5, EF, 3F, 1A, 4C, FE, FC

There are 3 voice frames in each voice burst. Let the bits in first voice frame have prefix of v1, that of second voice frame of v2 and that of third voice frame of v3 for clarity in representation.

Identifying the location of bits in one AMBE+2 encoded frame can be understood by going through the AMBE+2 encoding process itself.

The vocoder segments voice into 20 ms frames. Each 20 ms frame contains 49 speech bits. The digitized voice data (49 bits) is grouped into 4 vectors, denoted as u_0, u_1, u_2, and u_3. u_0 is the most significant vector and the u_3 is the least significant vector. Within each vector, the bits are numbered with bit-0 corresponding to the least significant and bit-n being more significant as the index n increases. For example, u_0(0) is the least significant bit and the u_0(11) is the most significant bit in vector u_0. The 12 most significant bits are contained in u_0 and it undergoes FEC encoding. The next 12 most significant bits are contained in u_1 and undergoes FEC encoding. Vectors u_2 (11 bits) and u_3 (14 bits) do not have any forward error correction code.

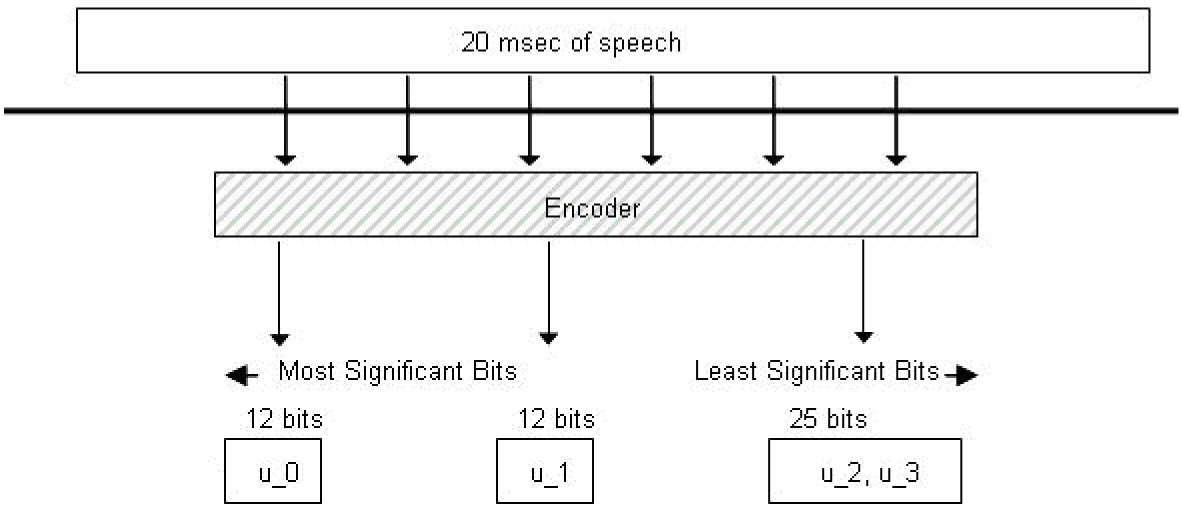


Figure 77: Voice code word construction

The diagram below shows how the AMBE vector bits are located in the AMBE Encoded Voice Frame field in the WL_VC_VOICE_BURST message.

Offset	Allocation							
	MSB 7	6	5	4	3	2	1	0 LSB
0	R	R	R	R	R	R	R	Bad Voice Frame
1	v1u_0(11) v1u_0(10) v1u_0(9) v1u_0(8) v1u_0(7) v1u_0(6) v1u_0(5) v1u_0(4)							
2	v1u_0(3) v1u_0(2) v1u_0(1) v1u_0(0) v1u_1(11) v1u_1(10) v1u_1(9) v1u_1(8)							
3	v1u_1(7) v1u_1(6) v1u_1(5) v1u_1(4) v1u_1(3) v1u_1(2) v1u_1(1) v1u_1(0)							
4	v1u_2(11) v1u_2(10) v1u_2(9) v1u_2(8) v1u_2(7) v1u_2(6) v1u_2(5) v1u_2(4)							
5	v1u_2(3) v1u_2(2) v1u_2(1) v1u_2(0) v1u_3(12) v1u_3(11) v1u_3(10) v1u_3(9)							
6	v1u_3(8) v1u_3(7) v1u_3(6) v1u_3(5) v1u_3(4) v1u_3(3) v1u_3(2) v1u_3(1)							
7	v1u_3(0)	Bad Voice Frame	v2u_0(11) v2u_0(10) v2u_0(9) v2u_0(8) v2u_0(7), v2u_0(6)					
8	v2u_0(5) v2u_0(4) v2u_0(3) v2u_0(2) v2u_0(1) v2u_0(0) v2u_1(11) v2u_1(10)							
9	v2u_1(9) v2u_1(8) v2u_1(7) v2u_1(6) v2u_1(5) v2u_1(4) v2u_1(3) v2u_1(2)							
10	v2u_1(1) v2u_1(0) v2u_2(11) v2u_2(10) v2u_2(9) v2u_2(8) v2u_2(7) v2u_2(6)							

11	v2u_2(5) v2u_2(4) v2u_2(3) v2u_2(2) v2u_2(1) v2u_2(0) v2u_3(12) v2u_3(11)		
12	v2u_3(10) v2u_3(9) v2u_3(8) v2u_3(7) v2u_3(6) v2u_3(5) v2u_3(4) v2u_3(3)		
13	v2u_3(2) v2u_3(1) v2u_3(0)	Bad Voice Frame	v3u_0(11) v3u_0(10) v3u_0(9) v3u_0(8)
14	v3u_0(7) v3u_0(6) v3u_0(5) v3u_0(4) v3u_0(3) v3u_0(2) v3u_0(1) v3u_0(0)		
15	v3u_1(11) v3u_1(10) v3u_1(9) v3u_1(8) v3u_1(7) v3u_1(6) v3u_1(5) v3u_1(4)		
16	v3u_1(3) v3u_1(2) v3u_1(1) v3u_1(0) v3u_2(11) v3u_2(10) v3u_2(9) v3u_2(8)		
17	v3u_2(7) v3u_2(6) v3u_2(5) v3u_2(4) v3u_2(3) v3u_2(2) v3u_2(1) v3u_2(0)		
18	v3u_3(12) v3u_3(11) v3u_3(10) v3u_3(9) v3u_3(8) v3u_3(7) v3u_3(6) v3u_3(5)		
19	v3u_3(4) v3u_3(3) v3u_3(2) v3u_3(1) v3u_3(0)	R (bits 2-0)	

Figure 78: AMBE Voice Vector Bit Lcation in AMBE Encoded Voice Frame Field

The 72-bit FEC encoded IV bits, named as G71 (MSB) through G0 (LSB), are inserted in one voice superframe by stealing u_3(3), u_3(2), u_3(1), u_3(0) (4 least significant bits) in every 20-ms voice frame.The bit stealing and IV insertion table for one superframe voice frames is shown in Table 30.

IV bit index	Frame ID	Bit Name	IV bit index	Frame ID	Bit Name	IV bit index	Frame ID	Bit Name
G68	A	v1u_3(3)	G60	C	v1u_3(3)	G52	E	v1u_3(3)
G69		v1u_3(2)	G61		v1u_3(2)	G53		v1u_3(2)
G70		v1u_3(1)	G62		v1u_3(1)	G54		v1u_3(1)
G71		v1u_3(0)	G63		v1u_3(0)	G55		v1u_3(0)
G44		v2u_3(3)	G36		v2u_3(3)	G28		v2u_3(3)
G45		v2u_3(2)	G37		v2u_3(2)	G29		v2u_3(2)
G46		v2u_3(1)	G38		v2u_3(1)	G30		v2u_3(1)
G47		v2u_3(0)	G39		v2u_3(0)	G31		v2u_3(0)
G20		v3u_3(3)	G12		v3u_3(3)	G4		v3u_3(3)
G21		v3u_3(2)	G13		v3u_3(2)	G5		v3u_3(2)
G22		v3u_3(1)	G14		v3u_3(1)	G6		v3u_3(1)
G23		v3u_3(0)	G15		v3u_3(0)	G7		v3u_3(0)
G64	B	v1u_3(3)	G56	D	v1u_3(3)	G48	F	v1u_3(3)
G65		v1u_3(2)	G57		v1u_3(2)	G49		v1u_3(2)
G66		v1u_3(1)	G58		v1u_3(1)	G50		v1u_3(1)
G67		v1u_3(0)	G59		v1u_3(0)	G51		v1u_3(0)
G40		v2u_3(3)	G32		v2u_3(3)	G24		v2u_3(3)
G41		v2u_3(2)	G33		v2u_3(2)	G25		v2u_3(2)
G42		v2u_3(1)	G34		v2u_3(1)	G26		v2u_3(1)
G43		v2u_3(0)	G35		v2u_3(0)	G27		v2u_3(0)
G16		v3u_3(3)	G8		v3u_3(3)	G0		v3u_3(3)
G17		v3u_3(2)	G9		v3u_3(2)	G1		v3u_3(2)
G18		v3u_3(1)	G10		v3u_3(1)	G2		v3u_3(1)
G19		v3u_3(0)	G11		v3u_3(0)	G3		v3u_3(0)

Table 30: Substitution of IV in voice superframe

'G68, G69, G70 and G71' IV bits are inserted in the least significant 4 bits of the first voice frame of burst A which are the locations for v1u_3(3), v1u_3(2), v1u_3(1) and v1u(0) in . Note that in this context a voice frame is referred to as 49-bits of AMBE+2 encoded frame. G68 is substituted for v1u_3(3), G69 is substituted for v1u_3(2), G70 is substituted for v1u_3(1) and G71 is substituted for v1u_3(0). Similarly, 'G44, G45, G46 and G47' IV bits are inserted in the least significant 4 bits of second voice frame of burst A and so on for the entire superframe.

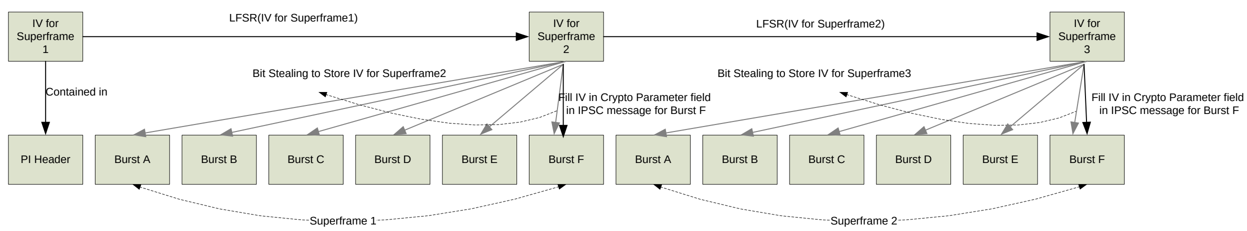


Figure 79: IV Storage in the Voice Stream

As shown in the figure above, when a third party application initiates a voice call with enhanced privacy, the IV for superframe1 shall be put in the WL_VC_PRIVACY_BURST, which corresponse to the over-the-air PI header, the IV for superframe2 shall be stored in the superframe1 by bit stealing the audio bits. The third party application shall also fill the IV for superframe2 in the WL_VC_VOICE_BURST message for Burst F.

5 System Configuration

The Linked Establishment ADK (Reference [1]) describes the system configurations that are needed to in the repeater and third party application for the link establishment. This section only covers the configuration for the Call Control interface.

5.1 Linked Capacity Plus / Capacity Plus Rest Channel IP Address / UDP Port

In both Linked Capacity Plus and Capacity Plus system, a fixed IP address and a fixed UDP port are assigned to the Rest Channel. This static IP address and UDP port of the Rest Channel is associated with a repeater only for the duration for which one of its logical channels is the Rest Channel. Every repeater in the same Capacity Plus system has to configure the same Rest Channel IP address and port number. Every repeater in the same site of the Linked Capacity Plus system has to configure the same Rest Channel IP address and port number. The third party application can get the Rest Channel IP address and port number from the LE system map.

In Linked Capacity Plus system, the Data Revert repeaters shall also configure the Rest Channel IP address and port number.

Figure 80 shows the Rest Channel IP/UDP Port Configuration.

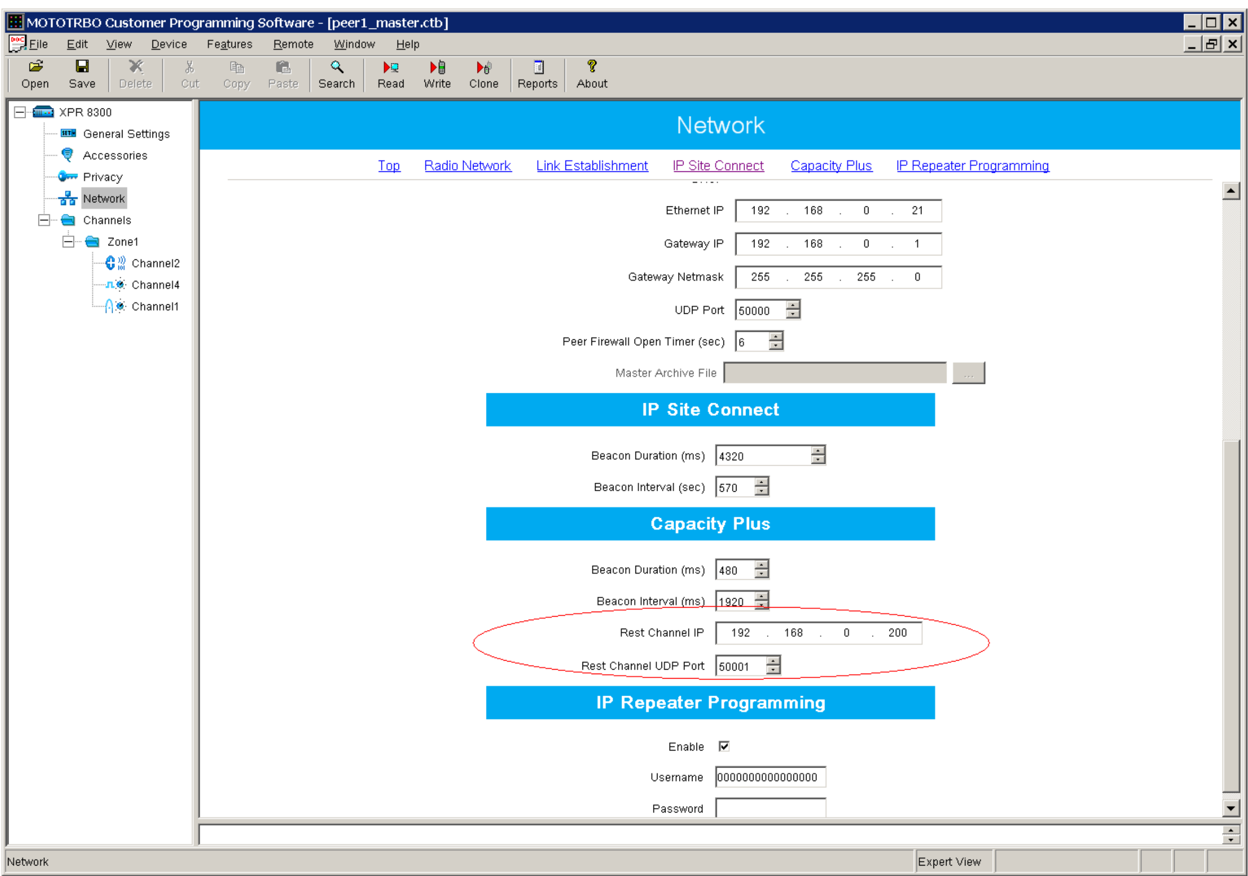


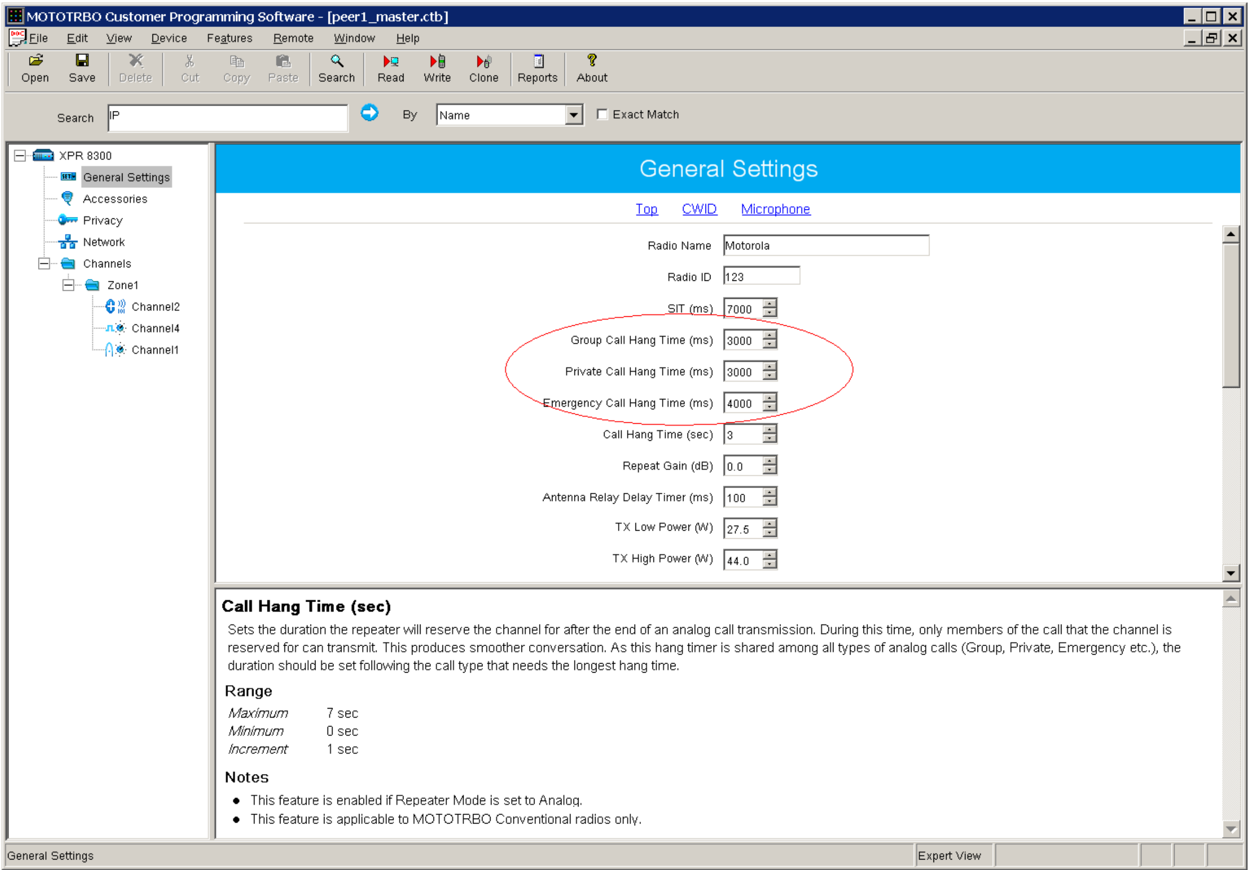
Figure 80: Rest Channel IP/UDP Port Configuration

5.2 Call Hang Time Configuration

As specified in section 3.5.1 and 3.6.1, at the end of a voice/CSBK transmission, the MOTOTRBO repeater will waits for the Call Hang Time duration before ending the session. There are different call hang times for voice and CSBK transmission. The Call Hang Time value for the transmission types are shown in the table below. The application shall send the CSBK ACK within 300ms after receiving a CSBK command from the radio.

<table>
<tr><th>Transmission Type</th><th>Call Hang Time Value (Seconds)</th><th>Comments</th></tr>
<tr><td>Group Voice Call</td><td>3 (by default)</td><td>It is CPS configurable. 3 seconds by default.</td></tr>
<tr><td>Private Voice Call</td><td>4 (by default)</td><td>It is CPS configurable. 4 seconds by default.</td></tr>
<tr><td>Emergency Voice Call</td><td>4 (by default)</td><td>It is CPS configurable. 4 seconds by default.</td></tr>
<tr><td>All System Call</td><td>0</td><td>No Call Hang Time.</td></tr>
<tr><td>CSBKs</td><td rowspan="2">0.87(Shortest value)</td><td rowspan="2"></td></tr>
<tr><td>ACK for OACSU / Emergency Alarm / Remote Monitor CSBK</td></tr>
<tr><td>ACK for Call Alert / Extended Function Command and the Rest of the CSBK ACKs</td><td>0.54(Shortest value)</td><td></td></tr>
</table>

For proper functioning of the repeater system, all the repeater peers in the same system should have the same call hang times. Figure 81 shows the Call Hang Time configuration for Group Voice call, Private Voice call, and Emergency Voice call.



2435

2436 **Figure 81: Call Hang Time Configuration**

5.3 *Third Party Application Configuration List*

This section summarizes the necessary configuration items for a third party application that needs to process voice/CSBK call or call recording, it should provide the following configuration items:

- Peer ID
- Master IP Address
- Master UDP Port
- Application peer's IP address
- Application peer's UDP port
- Default Gateway
- Authentication Key (Optional, must be identical to the repeater peers)
- Wireline Authentication Vendor ID (Get from Motorola ADP regional manager, and must be protected in the application)
- Wireline Authentication Key (Get from Motorola ADP regional manager, and must be protected in the third party application)
- Firewall Open Timer
- Data Revert Slot (Only for Single Site or IP Site Connect System)
- Emergency Revert Slot
- Subscriber ID – In case the third party application operates as a Subscriber to initiate a voice/data/control call in the radio system.
- Individual / Group Call ID – to specify the target radio in a voice / CSBK call

6 Call Control Protocol Definition

The section is to define the Network Application Interface Call Control Protocol Message that is used by the 3rd party application in the repeater system. The Call Control Protocol is supported by the MOTOTRBO repeaters based on MOTOTRBO 2.2 release and higher, regardless of which mode the repeater operates on. In the sections below, the terms of Call Control and Wireline have the same meaning.

6.1 Call Control PDU Structure

Call Control Message works upon connectionless UDP/IP protocol. The Message structure defined figure below.

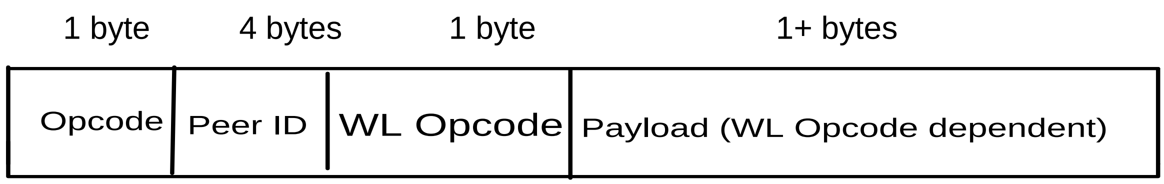
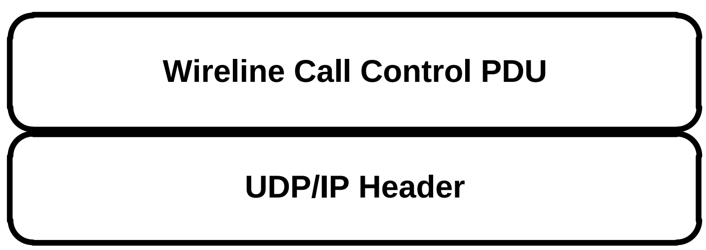


Figure 82 Call Control Protocol Message Format

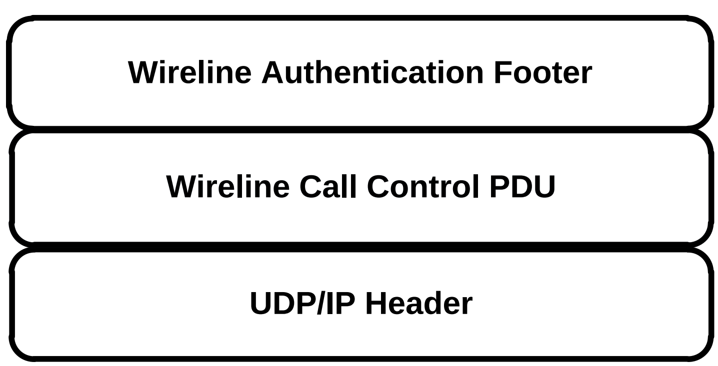
All Call Control messages have a one-byte opcode (0xB2), a four byte ID of the peer that is sending the message, and a one-byte Wireline opcode that specifies the type of the message. The payload is entirely dependent on the WL opcode.

6.2 Wireline Authentication Footer

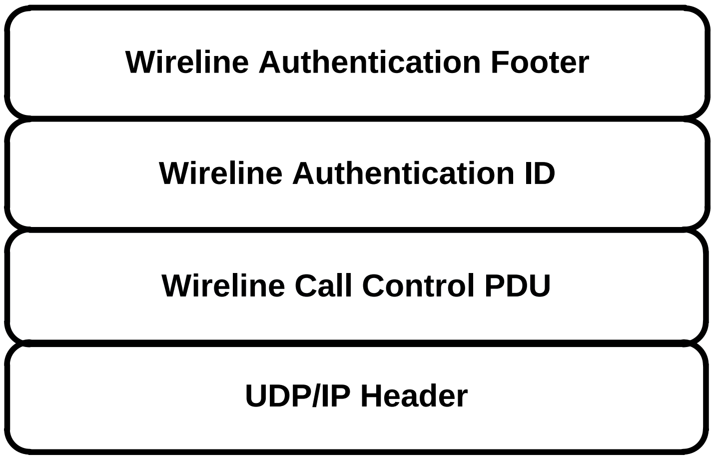
In the repeater peer, it is optional to configure the Authentication Key. If the Authentication Key is not configured, all the Call Control messages originated from the repeater peer do not have the authentication footer as shown in the diagram below:



If the Authentication Key is configured, at the end of all the Call Control messages originated from the repeater peer have the authentication footer as shown in the diagram below:



See Reference [1] for more details on the Authentication Footer Calculation.
It is optional for the third party application to configure the Authentication Key. However, no matter the Authentication Key is configured or not, at the end of all Call Control messages originated from the third party application, it is required to have the Wireline Authentication Footer as shown in the diagram below:



See section 3.4 for details on the Wireline Authentication Footer calculation.

6.3 Message Direction

The Call control Protocol supports 3 message directions:

- **A → R** From Third Party Application to Repeater Peer
- **R → A** Form Repeater Peer to Third Party Application
- **A ↔ R** Between Third Party Application and Repeater Peer

6.4 Byte Order

All Byte defined in this ADK is Big Endian unless otherwise specified

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

6.5 Wireline Opcode

Wireline Opcodes or Call Control Opcodes are defined in table below table. For compatibility reason, the third party application shall discard any message with unknown Opcode embedded.

Opcode	Value
WL_REGISTRATION_REQUEST	0x01
WL_REGISTRATION_STATUS	0x02
WL_REGISTRATION_GENERAL_OPS	0x03
WL_CHNL_STATUS	0x11

WL_CHNL_STATUS_QUERY	0x12
WL_VC_CHNL_CTRL_REQUEST	0x13
WL_VC_CHNL_CTRL_STATUS	0x16
WL_VC_CSBK_CALL	0x17
WL_VC_VOICE_START	0x18
WL_VC_VOICE_END_BURST	0x19
WL_VC_CALL_SESSION_STATUS	0x20
WL_VC_VOICE_BURST	0x21
WL_VC_PRIVACY_BURST	0x22

Table 31 – Wireline Opcode

6.6 Key to message Specification

This section defines the Call Control message specification template that is used to describe each command.

6.6.1 Message Dashboard

Class	Call Control	Type	Request
Opcode	0x01	Command	WL_REGISTRATION_REQUEST
Description	Wireline Registration Request		

At the top of each specification, there is a dashboard depicting the key characteristics of the Call Control message. The sections of the dashboard are described below:

- Class – Category of the messages that share common properties, operations.
- Direction – Indicates whether the message is send from the third party application to Repeater Peer, from Repeater Peer to the third party application, or bidirectional.
- Opcode – Static enumerated value assigned to the message; the size of this value is 1 byte.
- Command – Common alphabetic alias for the message.
- Description – Elaborated definition of the Command assigned to the message.

6.6.2 Message Field Types

Uint8 – A 8-bit unsigned integer.

Uint16 - A 16-bit unsigned integer.

Uint24 – A 24-bit unsigned integer.

Uint32 - A 32-bit unsigned integer.

String – A NULL terminated array of UCS-2 Unicode characters, unless otherwise specified in the message

6.6.3 Reserved Fields

Some Call Control messages may have reserved fields identified in the message structure. These fields have been identified for future use and should not be utilized in any way. For any fields marked as "(Reserved)", the value assigned to that field must be 0x00 up to the length / size of the reserved field. Failure to do so may result in unexpected operation or behavior.

6.6.4 Packet Format per Call Control Protocol Version

System	Version Introduced
IP Site Connect	1

At the top of each packet format table is an overhead table identifying its Call Control protocol version.

The Call Control protocol version has two fields: system ID and protocol version. The MOTOTRBO repeater can only support the current link protocol version. The third party application shall follow the same backward compatibility depth as the MOTOTRBO repeater peer.

- Version Introduced – The protocol version at which the packet format starts. The Version Introduced is the version protocol field in the Call Control protocol version. If the packet format changes, a separate message format table shows the new format in the new protocol version.

- System – The system at which the packet format is supported. It is the system ID in the Call Control protocol version.

6.7 Call Control Message Definition

6.7.1 Basic Message Format

The basic structure for a Call Control Message is shown below.

Field	Type	Description
opcode	Uint8	0xB2
peerID	Uint32	The ID of the sending peer
Wireline Opcode	Uint8	Specifies the OP Code of the PDU
Wireline Opcode Specific Field 1		
……..		
……..		
Wireline Opcode Specific Field N		
Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version
Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version

Note: In each specific packet format table, there is a heading described this message is introduced from which system and version. For more details, refer to section 6.6.4.

6.7.2 0x01– WL_REGISTRATION_REQUEST

Class	Call Control	Direction	A → R
Opcode	0x01	Command	WL_REGISTRATION_REQUEST
Description	Wireline Registration Request		

6.7.2.1 Description

This message is sent from the third party application to the repeater peers to subscribe for calls that are of its interest.

A third party application sends out a Wireline Registration Request in one of the following situations:

1) Third party application establishes link with the repeaters peer by Link Establishment protocol.

The repeater peer wills route the received call to the third party application if it is subscribe. There can be call multiple subscriptions in the Wireline Registration Request. The registration request can be base on radio addresses or talk group addresses.

Wireline Registration Request allowed having maxim 16 registration entries for radio addresses and 16 registration entries for talk group addresses.

6.7.2.2 Cautions / Warnings

None

6.7.2.3 Packet Format

System	Version Introduced
Single Site/IP Site Connect	1
Capacity Plus	1
Linked Capacity Plus	1

Offset	Field	Type	Description	Information Field
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	6.8.1
5	wirelineOpcode	Uint8	Value = 0x01	
6	registrationSlotNumber	Uint8	Only for SS/IPSC mode Specifies the slot number for register.	6.8.2
7	registrationPduID	Uint32	The identifier of the registration PDU.	6.8.3
11	registrationID	Uint16	The identifier associated with the registration profile in this PDU.	6.8.4
13	wirelineStatusRegistration	Uint8	Registration flag for Wireline Channel Status	6.8.5
14	numberOfRegistrationEntries	Uint8	Number of registration entries in this registration message.	

System		Version Introduced		
Single Site/IP Site Connect		1		
Capacity Plus		1		
Linked Capacity Plus		1		
Offset	**Field**	**Type**	**Description**	**Information Field**
15	AddressType(1)	Uint8	Address type of the call being registered in 1^{st} entry	6.8.6
16	addressRangeStart(1)	Uint32	Start of radio/group addresses (CAI ID) range in 1^{st} entry.	6.8.7
20	addressRangeEnd(1)	Uint32	End of radio/group addresses (CAI ID) range in 1^{st} entry.	6.8.7
24	VoiceAttributes(1)	Uint8	Attributes of Voice Registration in 1^{st} entry.	6.8.8
25	CSBKAttributes(1)	Uint8	Attributes of CSBK Registration in 1^{st} entry	6.8.9
26	RESERVED(1)	Uint8		
	……..			
	……..			
(N-1)* 12+ 15	AddressType(N)	Uint8	Address type of the call being registered in N^{th} entry.	6.8.6
(N-1)* 12+ 16	addressRangeStart(N)	Uint32	Start of radio/group addresses (CAI ID) range in N^{th} entry.	6.8.7
(N-1)* 12+ 20	addressRangeEnd(N)	Uint32	End of radio/group addresses (CAI ID) range in N^{th} entry..	6.8.7
N-1)* 12+ 24	VoiceAttributes(N)	Uint8	Attributes of Voice Registration in N^{th} entry..	6.8.8
N-1)* 12+ 25	CSBKAttributes(N)	Uint8	Attributes of CSBK Registration in N^{th} entry.	6.8.9
(N-1)* 12+ 26	RESERVED(N)	Uint8		
(N-1)* 12+ 27	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
(N-1)* 12+ 28	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

6.7.3 0x02– WL_REGISTRATION_ GENERAL_OPS

Class	Call Control	Direction	A → R
Opcode	0x02	Command	WL_REGISTRATION_ GENERAL_OPS
Description	Wireline Registration General Operation		

6.7.3.1 Description

This message is sent from the third party application to the repeater peers to de-register its prior registration or query the repeater peer to find out the Identifier of the registration.

6.7.3.2 Cautions / Warnings

None

6.7.3.3 Packet Format

System		Version Introduced		
Single Site/IP Site Connect		1		
Capacity Plus		1		
Linked Capacity Plus		1		
Offset	**Field**	**Type**	**Description**	**Information Field**
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	6.8.1
5	wirelineOpcode	Uint8	Value = 0x03	
6	registrationSlotNumber	Uint8	Only for de-register operation in SS/IPSC mode Specifies the slot number for operation.	6.8.2
7	registrationPduID	Uint32	The identifier of the registration PDU.	6.8.3
11	registrationOperationOpcode	Uint8	The registration Operation Code	6.8.11
12	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
13	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

6.7.4 0x03– WL_REGISTRATION_ STATUS

Class	Call Control	Direction	R → A
Opcode	0x03	Command	WL_REGISTRATION_ STATUS
Description	Wireline Registration Status		

6.7.4.1 Description

This message is sent from the repeater peer to the third party applications to response the Wireline Registration Request or Wireline Registration General Operation.

A repeater peer sends out a Wireline Registration Status in one of the following situations:

1) Upon receiving a Wireline Registration Request from the third party application.
2) Upon receiving a Wireline Registration General Operation from the third party application.

6.7.4.2 Cautions / Warnings

None

6.7.4.3 Packet Format

System	Version Introduced
Single Site/IP Site Connect	1
Capacity Plus	1
Linked Capacity Plus	1

Offset	Field	Type	Description	Information Field
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	6.8.1
5	wirelineOpcode	Uint8	Value = 0x02	
6	registrationPduID	Uint32	The identifier of the registration PDU.	6.8.3
10	registrationID (Slot 1)	Uint16	The identifier associated with the registration profile in this PDU.	6.8.4
12	registrationID (Slot 2)	Uint16	The identifier associated with the registration profile in this PDU.	6.8.4
14	registrationStatus	Uint8	The registration status	6.8.12
15	registrationStatusCode	Uint8	The registration status code	6.8.13
16	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
17	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

6.7.5 WL_CHNL_STATUS

Class	Call Control	Direction	R → A
Opcode	0x011	Command	WL_ CHNL_STATUS
Description	Wireline Channel Status		

6.7.5.1 Description

This message is sent from the repeater peer to the third party applications to report the conventional channel status of SS/IPSC system or rest channel status of CPP/LCP system. This message is only sent out by the first repeater peer, which receives the call over the air from the radio or receives the call over the network from the third party application.

Only the third party application who has subscribed the wireline channel status during wireline registration can receive this message.

A repeater peer sends out a Wireline Channel Status in one of the following situations:

1) Upon receiving a Wireline Channel Status Query from the third party application.

2) The register channel status of SS/IPSC system or rest channel status of CPC/LCP system is updated.

In MOTOTRBO 2.2, the data revert repeaters in CPP/LCP system didn't send Wireline Channel Status PDU.

6.7.5.2 Cautions / Warnings

None

6.7.5.3 Packet Format

System	Version Introduced
Single Site/IP Site Connect	1
Capacity Plus	1
Linked Capacity Plus	1

Offset	Field	Type	Description	Information Field
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	6.8.1
5	wirelineOpcode	Uint8	Value = 0x11	
6	slotNumber	Uint8	Specifies the channel/slot reporting the channel status.	6.8.15
7	statusPduID	Uint32	The ID of the status PDU uniquely identifies the channel status PDU.	6.8.14
11	conventionalchannelStatus	Uint8	Only available in IPSC modes. The channel status information.	6.8.16

System		Version Introduced		
Single Site/IP Site Connect		1		
Capacity Plus		1		
Linked Capacity Plus		1		
Offset	Field	Type	Description	Information Field
12	restChannelStatus	Uint8	Only available in CPP/LCP modes. The rest channel status information.	6.8.16
13	typeOfCall	Uint8	Call type	6.8.20
14	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
15	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

6.7.6 0x12– WL_CHNL_STATUS_QUERY

Class	Call Control	Direction	A → R
Opcode	0x012	Command	WL_ CHNL_STATUS _QUERY
Description	Wireline Channel Status Query		

6.7.6.1 Description

This message is sent from the third party applications to the repeater peer to query the channel status.

The third party applications can send the WL_ CHNL_STATUS _QUERY to the:

- Repeater with the local area channel.
- Repeater with the wide area channel.
- Repeater with rest channel.

In MOTOTRBO R2.2 the CPP/LCP data revert repeaters are not required to send channel status.

6.7.6.2 Cautions / Warnings

None

6.7.6.3 Packet Format

System		Version Introduced		
Single Site/IP Site Connect		1		
Capacity Plus		1		
Linked Capacity Plus		1		
Offset	**Field**	**Type**	**Description**	**Information Field**
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	6.8.1
5	wirelineOpcode	Uint8	Value = 0x12	
6	slotNumber	Uint8	Specifies the channel/slot reporting the channel status.	6.8.15
7	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
8	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

6.7.7 0x13– WL_VC_CHNL_CTRL_REQUEST

Class	Call Control	Direction	A → R
Opcode	0x013	Command	WL_VC_CHNL_CTRL_REQUEST
Description	Wireline Voice Call Channel Control Request		

6.7.7.1 Description

This message is sent from the third party applications to the repeater peer to initial the voice call or send CSBK call request or response.

6.7.7.2 Cautions / Warnings

None

6.7.7.3 Packet Format

System		Version Introduced		
Single Site/IP Site Connect		1		
Capacity Plus		1		
Linked Capacity Plus		1		
Offset	**Field**	**Type**	**Description**	**Information Field**
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	6.8.1
5	wirelineOpcode	Uint8	Value = 0x13	
6	slotNumber	Uint8	Specifies the channel/slot	6.8.15
7	callID	Uint32	The id of call	6.8.17
11	callType	Unit8	The type of call	6.8.20
12	source ID	Uint32	The ID of the subscriber that initiated the call	6.8.18
16	target ID	Uint32	The ID of the subscriber or talk-group to which the call is targeted	6.8.19
20	accessCriteria	Uint8	The priority type to access the channel.	6.8.22
21	callAttributes	Uint8	General Call Attributes	6.8.23
22	RESERVED	Uint8		
23	preambleDuration	Uint8	Only available for Confirmed Private Voice/CSBK Call. The numbers of Preamble bursts send before transmission.	6.8.24
24	RESERVED	Uint16		
26	CSBK Arguments	Uint64	Only available for CSBK call The arguments encoded in the DMR CSBK burst.	6.8.25
34	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
35	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

6.7.8 0x16 – WL_VC_CHNL_CTRL_STATUS

Class	Call Control	Direction	R → A
Opcode	0x016	Command	WL_VC_CHNL_CTRL_STATUS
Description	Wireline Voice Call Channel Control Status		

6.7.8.1 Descriptions

This message is sent from the repeater peer to the third party applications to response the Wireline Voice Call Channel Control Request.

6.7.8.1 Cautions / Warnings

None

6.7.8.2 Packet Format

System		Version Introduced		
Single Site/IP Site Connect		1		
Capacity Plus		1		
Linked Capacity Plus		1		
Offset	**Field**	**Type**	**Description**	**Information Field**
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	6.8.1
5	wirelineOpcode	Uint8	Value = 0x16	
6	slotNumber	Uint8	Specifies the channel/slot	6.8.15
7	callID	Uint32	The id of call	6.8.17
11	callType	Unit8	The type of call.	6.8.20
12	chnCtrlstatus	Uint8	Channel control Status	6.8.26
13	DeclineReasonCode	Uint8	The reason for decline the call control request.	6.8.27
14	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
15	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

6.7.9 0x17 – WL_VC_CSBK_CALL

Class	Call Control	Direction	R → A
Opcode	0x017	Command	WL_VC_CSBK_CALL
Description	Wireline Voice Call CSBK CALL		

6.7.9.1 Description

This message is sent from the repeater peer to the third party application to send the CSBK request or response message.

6.7.9.2 Cautions / Warnings

None

6.7.9.3 Packet Format

System	Version Introduced
Single Site/IP Site Connect	1
Capacity Plus	1
Linked Capacity Plus	1

Offset	Field	Type	Description	Information Field
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	6.8.1
5	wirelineOpcode	Uint8	Value = 0x17	
6	slotNumber	Uint8	the channel/slot number	6.8.15
7	callID	Uint32	The id of call	6.8.17
11	callType	Unit8	The type of call.	6.8.20
12	source ID	Uint32	The ID of the subscriber that initiated the call	6.8.18
16	target ID	Uint32	The ID of the subscriber or talk-group to which the call is targeted	6.8.19
20	RESERVED	Uint8		
21	MFID	Uint8	The DMR Feature Set ID / Manufacturer's ID	6.8.28
22	CSBK Arguments	Uint64	Only available for CSBK call The arguments encoded in the DMR CSBK burst.	6.8.25
30	rawRssiValue	Uint16	The raw RSSI value for each burst.	6.8.33
32	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
33	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

6.7.10 0x18 –WL_VC_VOICE_START

Class	Call Control	Direction	R → A
Opcode	0x018	Command	WL_VC_VOICE_START
Description	Wireline Voice Call Start		

6.7.10.1 Description

This message is sent by the repeater peer to indicate the beginning of a voice stream.

6.7.10.2 Cautions / Warnings

None

6.7.10.3 Packet Format

System	Version Introduced
Single Site/IP Site Connect	1
Capacity Plus	1
Linked Capacity Plus	1

Offset	Field	Type	Description	Information Field
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	6.8.1
5	wirelineOpcode	Uint8	Value = 0x18	
6	slotNumber	Uint8	the channel/slot number	6.8.15
7	callID	Uint32	The id of call	6.8.17
11	callType	Unit8	The type of call.	6.8.20
12	source ID	Uint32	The ID of the subscriber that initiated the call	6.8.18
16	target ID	Uint32	The ID of the subscriber or talk-group to which the call is targeted	6.8.19
20	callAttributes	Uint8	Call General attributes	6.8.23
21	RESERVED	Uint8		
22	MFID	Uint8	The DMR Feature Set ID / Manufacturer's ID	6.8.28
23	serviceOption	Uint8	DMR LC Service Option	6.8.31
24	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
25	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

6.7.11 0x19 – WL_VC_VOICE_END_BURST

Class	Call Control	Direction	A ↔ R
Opcode	0x019	Command	WL_ VC_VOICE_END_BURST
Description	Wireline Voice Call End Burst		

6.7.11.1 Description

This message is sent by the repeater peer or the third party application to indicate the end of a voice stream.

6.7.11.2 Cautions / Warnings

None

6.7.11.3 Packet Format

System	Version Introduced
Single Site/IP Site Connect	1
Capacity Plus	1
Linked Capacity Plus	1

Offset	Field	Type	Description	Information Field
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	6.8.1
5	wirelineOpcode	Uint8	Value = 0x19	
6	slotNumber	Uint8	the channel/slot number	6.8.15
7	callID	Uint32	The id of call	6.8.17
11	callType	Unit8	The type of call.	6.8.20
12	source ID	Uint32	The ID of the subscriber that initiated the call	6.8.18
16	target ID	Uint32	The ID of the subscriber or talk-group to which the call is targeted	6.8.19
20	RTP Information Field	Uint8 * 12	The RTP structure information for associated voice stream.	6.8.29
32	burstType	Uint8	DMR voice burst type Must be Voice Terminator	6.8.30
33	RESERVED	Uint8		
34	MFID	Uint8	The DMR Feature Set ID / Manufacturer's ID	6.8.28
35	serviceOption	Uint8	Only for WL outbound DMR LC Service Option.	6.8.31
36	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
37	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

Note starting from R2.3, the repeater will not check or set the Current/Accepted Wireline Protocol Version and the Oldest Wireline Protocol Version in this message. The application shall not verify the Current/Accepted Wireline Protocol Version and the Oldest Wireline Protocol Version when receiving this message from the repeater peer.

6.7.12 0x20 – WL_VC_CALL_SESSION_STATUS

Class	Call Control	Direction	R → A
Opcode	0x020	Command	WL_ VC_CALL_SESSION_STATUS
Description	Wireline Voice Call Session Status		

6.7.12.1 Description

This message is sent from repeater peer to the third party application to inform the call session status. Only the repeater peer, which receives the call over the air from the radio or receives the call from the third party application over the network interface, sends out the WL_VC_CALL_SESSION_STATUS message.

6.7.12.2 Cautions / Warnings

None

6.7.12.3 Packet Format

System	Version Introduced
Single Site/IP Site Connect	1
Capacity Plus	1
Linked Capacity Plus	1

Offset	Field	Type	Description	Information Field
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	6.8.1
5	wirelineOpcode	Uint8	Value = 0x20	
6	slotNumber	Uint8	Specifies the channel/slot reporting the channel status.	6.8.15
7	callID	Uint32	The id of call for which this status is being reported.	6.8.17
11	callType	Unit8	The type of call	6.8.20
12	source ID	Uint32	The ID of the subscriber that initiated the call	6.8.18
16	target ID	Uint32	The ID of the subscriber or talk-group to which the call is targeted	6.8.19
20	RESERVED	Uint32		
24	callSessionStatus	Uint8	Call Session Stats.	6.8.21
25	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
26	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

6.7.13 0x21– WL_VC_VOICE_BURST

Class	Call Control	Direction	A ↔ R
Opcode	0x021	Command	WL_ VC_VOICE_BURST
Description	Wireline Voice Call Burst		

This message is used by the third party application or Repeater Peer to send Voice Burst.

6.7.13.1 Cautions / Warnings

None

6.7.13.2 Packet Format

System	Version Introduced
Single Site/IP Site Connect	1
Capacity Plus	1
Linked Capacity Plus	1

Offset	Field	Type	Description	Information Field
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	6.8.1
5	wirelineOpcode	Uint8	Value = 0x21	
6	slotNumber	Uint8	Specifies the channel/slot reporting the channel status.	6.8.15
7	callID	Uint32	The id of call	6.8.17
11	callType	Unit8	The type of call	6.8.20
12	source ID	Uint32	The ID of the subscriber that initiated the call	6.8.18
16	target ID	Uint32	The ID of the subscriber or talk-group to which the call is targeted	6.8.19
20	callAttributes	Uint8	Call General attributes	6.8.23
21	RESERVED	Uint8		
22	RTP Information Field	Uint8 * 12	RTP Header Information	6.8.29
34	RESERVED	Uint32		
38	burstType	Uint8	DMR Voice burst Type	6.8.30
39	RESERVED	Uint8		
40	MFID	Uint8	Only available for WL Outbound The DMR Feature Set ID / Manufacturer's ID	6.8.28
41	serviceOptions	Uint8	Only available for WL Outbound DMR LC Service Option	6.8.31
42	algorithmID	Uint8	Algorithm ID	6.8.34
43	keyID	Uint8	ID of the Privacy Key	6.8.34
44	IV	Uint32	Initialization Vector	6.8.34
48	AMBE voice encoded frames	Uint8 * 20	The AMBE voice encoded frames.	6.8.32

System		Version Introduced		
Single Site/IP Site Connect		1		
Capacity Plus		1		
Linked Capacity Plus		1		
Offset	**Field**	**Type**	**Description**	**Information Field**
68	rawRssiValue	Uint16	Only for WL outbound The raw RSSI value for each voice burst.	6.8.33
70	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
71	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

Note starting from R2.3, the repeater will not check or set the Current/Accepted Wireline Protocol Version and the Oldest Wireline Protocol Version in this message. The application shall not verify the Current/Accepted Wireline Protocol Version and the Oldest Wireline Protocol Version when receiving this message from the repeater peer.

6.7.14 0x22– WL_VC_PRIVACY_BURST

Class	Call Control	Direction	A ↔ R
Opcode	0x022	Command	WL_VC_PRIVACY_BURST
Description	Wireline Voice Call Privacy Burst		

This message is use by the third party application or Repeater Peer to send Privacy Voice Burst.

6.7.14.1 Cautions / Warnings

None

6.7.14.2 Packet Format

System	Version Introduced
Single Site/IP Site Connect	1
Capacity Plus	1
Linked Capacity Plus	1

Offset	Field	Type	Description	Information Field
0	Opcode	Uint8	Value = 0xB2	
1	peerID	Uint32	The ID of the sending peer	
5	wirelineOpcode	Uint8	Value = 0x22	
6	slotNumber	Uint8	Specifies the channel/slot reporting the channel status.	
7	callID	Uint32	The id of call	6.8.17
11	callType	Unit8	The type of call	6.8.20
12	source ID	Uint32	The ID of the subscriber that initiated the call	6.8.18
16	target ID	Uint32	The ID of the subscriber or talk-group to which the call is targeted	6.8.19
20	RTP Information Field	Uint8 * 12	RTP Header Information	6.8.29
32	burstType	Uint8	DMR Voice burst Type Must be Privacy Header	6.8.30
33	RESERVED	Uint8		
34	MFID	Uint8	Only for WL outbound The DMR Feature Set ID / Manufacturer's ID	6.8.28
35	algorithmID	Uint8	Algorithm ID	6.8.34
36	keyID	Uint8	ID of the Privacy Key	6.8.34
37	IV	Uint32	Initialization Vector	6.8.34
41	Current / Accepted Wireline Protocol Version	Uint8	Current or Accepted Wireline Protocol Version	6.8.10
42	Oldest Wireline Protocol Version	Uint8	Oldest Wireline Protocol Version	6.8.10

Note starting from R2.3, the repeater will not check or set the Current/Accepted Wireline Protocol Version and the Oldest Wireline Protocol Version in this message. The application shall not verify the Current/Accepted Wireline Protocol Version and the Oldest Wireline Protocol Version when receiving this message from the repeater peer.

6.8 Information Field Details

6.8.1 Peer ID

This field specifies the unique identity of a peer. It is a 32-bit unsigned integer in network byte order.

System	Version Introduced
IP Site Connect	0
Capacity Plus	0
Peer ID Value	**Allocation**
0x00000000	RESERVED
0x00000001 to 0x00FFFCDF	Valid Range
0x00FFFCE0 to 0xFFFFFFFF	RESERVED

Table 32 - Peer ID Allocation for IPSC and CPC System

System		Version Introduced		
Linked Capacity Plus		3		
Offset	**Field**	**Type**	**Value**	**Description**
0	siteID	Uint8	0x00: for application peer 0x01-0F: for repeater peer	The ID of the site where the sending peer situated.
1	peerID	Uint24	0x000000: RESERVED 0x000001 to 0xFFFCDF: Valid Range 0xFFFCE0 to 0xFFFFFF: RESERVED	The ID of the sending peer

Table 33 – Peer ID Allocation for Linked Capacity plus System

6.8.2 Registration Slot Number

This field indicates the slot number for registration.

Value	Allocation
0x00	RESERVED
0x01	Slot 1
0x02	Slot 2
0x03	Both Slot 1 and 2
0x04-0xFF	RESERVED

Table 34: Slot number Allocation.

6.8.3 Registration PDU ID

This field indicates the registration PDU ID. It is a 32-bit unsigned integer in network byte order.

Registration PDU ID is used by the third party application to track status of multiple commands issued to the same Repeater Peer. The third party application will generate a Registration PDU ID in WL_REGISTRATION_REQUEST or WL_REGISTRATION_GENERAL_OPS. The Repeater Peer will respond the

WL_REGISTRATION_STATUS with the same Registration PDU ID to the third party application.

Value	Description
0x01--0xFFFFFFFF	Identification value set by the third party application

Table 35: Registration PDU ID Allocation.

6.8.4 Registration ID

This field indicates the registration ID associated with the registration profile in this PDU. It is a 16-bit unsigned integer in network byte order. The third party application will generate a Registration ID in WL_REGISTRATION_REQUEST and use the registration ID when query Wireline Registration or cancel Wireline Registration.

Value	Description
0x01--0xFFFF	Identification value set by the third party application

Table 36: Registration ID Allocation.

6.8.5 Wireline Channel Status Flag

This field indicates the register flag for Wireline Channel Status. If third party application set the flag to 1, the repeater peer will broadcast the Wireline Channel Status when channel status is change.

Bit	Description
0-6	RESERVED
7	0: Not Register WL channel status 1: Register WL channel status

Table 37: Wireline Channel Status Flag Allocation.

6.8.6 Address Type

This field indicates the address type of the call that is being registered.

Value	Allocation	Supported System Mode
0x00	RESERVED	RESERVED
0x01	Individual call	Single Site, IPSC, Capacity Plus, Linked Capacity Plus
0x02	Group call	Single Site, IPSC, Capacity Plus, Linked Capacity Plus
0x03	All Individual call	Single Site, IPSC, Capacity Plus, Linked Capacity Plus
0x04	All talkgroup call	Single Site, IPSC, Capacity Plus, Linked Capacity Plus
0x05	All Wide talkgroups call	Linked Capacity Plus
0x06	All Local talkgroups call	Linked Capacity Plus
0x07-0xFF	RESERVED	RESERVED

Table 38: The Address Type Allocation

6.8.7 Start/End of Register Addresses

This field indicates the Start/End of register Address. Depending on the Address Type, it is a group id or individual Subscriber id. If the Address Type is either "All Individual Call", "All Group Call", "All Wide Talkgroup Call", or "All Local Talkgroup Call", the Start/End of Register Addresses is ignored by the repeater.

6.8.8 Voice Attributes

This field indicates the Attributes of voice registration.

Bit	Description
0-5	RESERVED
0-5	RESERVED
6	0 : Normal Service 1 : Voice Monitor Only Voice Monitor Only shall only be set if the call type is individual call
7	0 : Not Registered Voice Service 1: Registered Voice Service

Table 39: The Voice Attributes Allocation

The Bit 7 shall only be set to 0 if this registration entry does not send/receive any voice call. If Bit 6 is set to "Voice Monitor Only", Bit 7 has to set to "Registered Voice Service" so that the application can receive the voice call.

6.8.9 CSBK Attributes

This field indicates the Attributes of CSBK registration.

Bit	Description
0-5	RESERVED
6	0: CSBK Service 1: CSBK Monitor Only CSBK Monitor Only shall only be set if the call type is individual call
7	0: Not Registered CSBK Service 1: Registered CSBK Service

Table 40: The CSBK Attributes Allocation

The Bit 7 shall only be set to 0 if this registration entry does not send/receive any CSBK call. If Bit 6 is set to "CSBK Monitor Only", Bit 7 has to set to "Registered CSBK Service" so that the application can receive the CSBK call.

6.8.10 Wireline Protocol Version

This field indicates the Wireline Protocol Version.

Bit	Description
Bit 0-1	RESERVED

Bit 2-7	Major version information.

Table 41: The Wireline Protocol Version Allocation

6.8.11 Registration Operation Code

This field indicates the Registration Operation Code.

Value	Description
0x01	Query the registration status
0x02	De-register the prior registration

Table 42: Registration Operation Code Allocation

6.8.12 Registration Status

This field indicates the registration status.

Value	Description
0x00	registration is successful
0x01	registration is unsuccessful

Table 43: registration status Allocation

6.8.13 Registration Status Code

This field indicates registration status code. It is only available when registration status is unsuccessful.

Value	Description
0x00	RESERVED
0x01	CFS is not enabled
0x02	Number of Registration Entries is greater than the max limit.
0x03	Incompatible Version

Table 44: registration status code Allocation

6.8.14 Status PDU ID

This field indicates the PDU ID of Wireline Channel Status message. It is a 32-bit unsigned integer in network byte order.

This value should be incremented for each WL_CHNL_STATUS Message that is sent.

Value	Description
0x00--0xFFFFFFFF	Identification value set by Repeater Status PDU ID = Last Status PDU ID + 1

Table 45: Status PDU ID Allocation.

6.8.15 Slot number

This field indicates the slot number used for Call Control PDU.

Value\System	SS/IPSC	CPC	LCP
0x00	RESERVED	Rest Channel	Site IP address
0x01	Slot 1	Slot 1	Slot 1
0x02	Slot 2	Slot 2	Slot 2
0x03-0xFF	RESERVED	RESERVED	RESERVED

Table 46: Slot number Allocation.

6.8.16 Channel Status

This field indicates the repeater channel status information.

Type Value	Allocation	Description
0x00	Reserved	
0x01	Active (Repeat or Call Hang)	Only for SS/IPSC system: The repeater where the call session starts reports this status. The repeater reports this event after floor arbitration.
0x02	Idle (Channel Hang, Reactivation, or Hibernating)	Only for SS/IPSC system: The repeater that reports Active Repeat reports this status.
0x03-0x09	RESERVED	
0x0A	Slot is blocked	Only for SS/IPSC system: The repeater reports this event when it has Signal Interference or when it starts CWID. Note: separate messages are sent for Slot 1&2.
0x0B	Slot is unblocked	Only for SS/IPSC system: Reported by the repeater that reports Slot is blocked when the blocking condition is removed.
0x0C-0x0F	RESERVED	
0x10	Busy Rest Channel (All Channels Busy)	Only for CPP/LCP system: Reported by the rest channel repeater at the site.
0x11	Rest Channel is idle/available	Only for CPP/LCP system: Reported by the repeater that reported Busy Rest Channel. The status is reported when the rest channel becomes Idle or the repeater no longer has the rest channel. Or Reported by the repeater that reported Rest Channel is blocked. This status is reported when the blocking condition is removed or the repeater is no longer the rest channel.

0x12	Local Group Calls Not Allowed	Only for LCP system Reported by the rest channel repeater at the site.
0x13	Local Group Calls Allowed	Only for LCP system Reported by the repeater that reported Local Group Calls Not Allowed.
0x014	Rest Channel is blocked	Only for CPP/LCP system: Report by the rest channel repeater when it has Signal Interference or it starts CWID and the rest channel cannot be assigned to an alternate repeater.
0x15-0xFF	RESERVED	

Table 47: Channel Status Information Allocation.

6.8.17 Call ID

This field indicates the Call ID. It is a 32 bit unsigned integer in network byte order.

Value	Allocation
0x00	Reserved
0x01--0xFFFFFFFF	Valid ranges.

Table 48: Call ID Allocation.

6.8.18 Source ID

This field indicates the source identity of a subscriber. It is a 24 bit unsigned integer in network byte order.

Value	Allocation
0x00	Reserved
0x01-- 0xFFFCDF	Valid ranges.

Table 49: Source ID Allocation.

6.8.19 Target ID

This field indicates the target identity of a subscriber. It is a 24 bit unsigned integer in network byte order.

Value	Allocation
0x00	Reserved
0x01-- 0xFFFCDF	Valid ranges.

Table 50: Target ID Allocation.

6.8.20 Call Type

This field indicates the Call type information.

Call Type Value	Allocation	System = IP Site Connect/ Capacity Plus	System = Linked Capacity Plus

Call Type Value	Allocation	System = IP Site Connect/ Capacity Plus	System = Linked Capacity Plus
0x00-0x31	RESERVED		
0x32	Preamble Private CSBK Call	X	
0x33	Preamble Group CSBK Call	X	
0x34	Preamble Emergency Call	X	
0x35-0x3F	RESERVED		
0x40	Emergency CSBK Alarm Request	X	X
0x41	Emergency CSBK Alarm Response	X	X
0x42	Emergency Voice Call	X	X
0x43	Private Call Request	X	X
0x44	Private Call Response	X	X
0x45	Call Alert Request	X	X
0x46	Call Alert Response	X	X
0x47	Radio Check Request (Extended Function Command)	X	X
0x48	Radio Check Response (Extended Function Response)	X	X
0x49	Radio Inhibit Request (Extended Function Command)	X	X
0x4A	Radio Inhibit Response (Extended Function Response)	X	X
0x4B	Radio Un-inhibit Request (Extended Function Command)	X	X
0x4C	Radio Un-inhibit Response (Extended Function Response)	X	X
0x4D	Radio Monitor Request	X	X
0x4E	Radio Monitor Response	X	X
0x4F	Group Voice Call	X	X
0x50	Private Voice Call	X	X
0x51-52	RESERVED		
0x53	All Call	X	X
0x54	RESERVED	X	X
0x55	Other Calls	X	X
0x56	IP Console Radio Un-Inhibit Request (Extended Function Command)	X	
0x57	IP Console Radio Inhibit Request (Extended Function Command)	X	

Call Type Value	Allocation	System = IP Site Connect/ Capacity Plus	System = Linked Capacity Plus
0x58	IP Console Radio Un-Inhibit Response (Extended Function Response)	x	
0x59	IP Console Radio Inhibit Response (Extended Function Response)	x	
0x5A	Group Phone Call	x	x
0x5B	Private Phone Call	x	x
0x5C	Phone All Call	x	x
0x83	Call Alert Nack Response	x	x
0x84	Radio Monitor Nack Response	x	x
0x85	Radio Inhibit/Un-inhibit Nack Response	x	x
0x86-0x8A	RESERVED		
0x8B	Wireline Remote Voice Dekey	x	x
0x8C-0xFF	RESERVED		

Table 51: Call type Allocation.

Notes:

- The third party application can receive the Private and Group phone calls but cannot initiate or participate in a Private or Group phone call.
- When call type is Remote Voice Dekey, only source ID is required in the call request PDU. The target ID must be set to 0x000000.
- Radio Inhibit and Un-inihibit Nack response are combined to one call type - Radio Inhibit/Un-inhibit Nack Response.

6.8.21 Call Session Status

This field indicates the Call Session Status.

Value	Allocation
0x00-0x09	RESERVED
0x0A	Call Session - Call Hang
0x0B	Call Session - End (channel hang)
0x0C-0xFF	RESERVED

Table 52: Call Session Status Allocation.

6.8.22 Access Criteria

This field indicates the Access Criteria Information of the call.

Value	Allocation
0x00	RESERVED

0x01	Polite Access
0x02	Transmit Interrupt
0x03	Impolite Access (Only for Voice Call)
0x04-0xFF	RESERVED

Table 53: Access Criteria Allocation.

6.8.23 Call Attributes

This field indicates the Call Attributes information.

Bit	Description
0	Only available for voice call initialize by repeater peer. 0: The voice call is not interruptible 1: The voice call is interruptible.
1	0: Clear Call 1: Privacy Call
2-3	0: Clear Call 1: Basic Privacy 2: Enhanced Privacy 3: Reserved
4	Reserved
5	Only available for private call in LCP mode. 0: the private voice or private CSBK call is forwarded to remote sites. 1: the private voice or private CSBK call is not forwarded to remote sites.
6-7	RESERVED

Table 54: Call Attributes Allocation

6.8.24 Preamble Duration

This field indicates the number of Preamble bursts send before transmission. The Preamble burst duration is 60ms. The minimum Preamble duration is 0ms and the maximum Preamble duration is 7680ms.

Value	Allocation
0x00 – 0x80	number of Preamble bursts (0ms – 7680ms)
0x81 – 0xFF	RESERVED

Table 55: Preamble Duration Allocation.

6.8.25 CSBK Arguments

This field indicates CSBK Argument information. Please refer the 7.1 for detailed information.

6.8.26 Channel Control Request Status

This field indicates the Status information for channel control request.

Value	Allocation	Description
0x00	RESERVED	
0x01	Received	The channel control request is received
0x02	Transmitting	The preamble CSBK burst is being transmitted.

0x03	Transmission Successful	The CSBK burst transmitted successfully.
0x04	Grant	The voice call was setup successfully and WL peer can send Voice bursts.
0x05	Declined	The repeater peer is Unable to Setup the call or Continue the call requested.
0x06	Interrupting	The Transmit Interrupt sequence is initiated.
0x07-0xFF	RESERVED	

Table 56: Channel Control Request Status.

6.8.27 Decline Reason Code

When a third party application fails to initiate a call or talk back in a Call Session at the Call Control interface using the WL_VC_CHNL_CTRL_REQUEST message, the repeater peer sends a WL_VC_CHNL_CONTROL_STATUS to indicate the result. The ChnlCtrlStatus field is set to decline, and the Reason Code field indicates the failure reason. Table 57 provides recommendation to third party application based on the failure reason.

Value	Reason Code	Failure Scenarios	CPC/LCP		SS/IPSC
			New Call On Rest Channel	Callback during call hang (in call session)	New Call or Callback during call hang
0x03	Race Condition Failure	Call Setup request is rejected during Arbitration.	The third party application waits for 90ms then auto-retries*. *The retry to access the new rest channel.* **The third party application resends the WL Call Setup request automatically, without the user pressing the PTT again.*	The third party application may indicate to the user that call request was unsuccessful or It may retry with access criteria to takeover the on-going call.	The third party application may indicate to the user that call request was unsuccessful or It may retry with access criteria to takeover the on-going call.
0x05	Destination Slot Busy Failure	The channel which the third party application is accessing is busy.	The third party application waits for 90ms then auto-retries. *The retry to access the new rest channel.*	The third party application may indicate to the user that call request was unsuccessful or It may retry with access criteria to takeover the on-going call.	The third party application may indicate to the user that call request was unsuccessful or It may retry with access criteria to takeover the on-going call.

Value	Reason Code	Failure Scenarios	CPC/LCP		SS/IPSC
			New Call On Rest Channel	Callback during call hang (in call session)	New Call or Callback during call hang
0x06	Destination Group Busy Failure	WL Call Setup request is declined because the destination Group is busy on another channel.	The third party application may indicate to the user that call request was unsuccessful.	n/a	n/a
0x07	All Channels Busy Failure	WL Call Setup request is declined because all the channels at the site are busy. The rest channel is busy.	The third party application may indicate to the user that call request was unsuccessful or It may retry with access criteria to takeover the on-going call.	The third party application may indicate to the user that call request was unsuccessful or It may retry with access criteria to takeover the on-going call.	n/a
0x08	OTA Repeat Disabled Failure	WL Call Setup request is declined because repeater where the request is sent is momentarily disabled by a system monitoring application	The third party application may indicate to the user that call request was unsuccessful. *The third party applications call is no longer being transmitted.*		
0x09	Signal Interference Failure	WL Call Setup request is declined because repeater where the request is sent is suffering FCC type I or II interference.	The third party application may indicate to the user that call request was unsuccessful.	n/a	The third party application may indicate to the user that call request was unsuccessful. Optionally, in case of IPSC wide area channel, the third party application

Value	Reason Code	Failure Scenarios	CPC/LCP		SS/IPSC
			New Call On Rest Channel	Callback during call hang (in call session)	New Call or Callback during call hang
					may choose another repeater to setup the call. The call will get transmitted at all the repeaters of the wide area channel, except the repeater where interference is occurring.
0x0A	CWID In Progress Failure	The WL Call Setup request is declined because the repeater where the request is sent is transmitting CWID.	The third party application may indicate to the user that call request was unsuccessful.	n/a	The third party application may indicate to the user that call request was unsuccessful. Optionally, in case IPSC wide area channel, the third party application may choose another repeater to setup the call. The call will get transmitted at all the repeaters of the wide area channel, except the repeater where CWID is being transmitted.
0x0B	TOT Expiry Premature Call End Failure	The call sending by the third party application is ended because of the TOT timer expiry.	The third party application indicates to the user that the call has ended. *The third party applications call is no longer being transmitted.*		

<table>
<tr><th rowspan="2">Value</th><th rowspan="2">Reason Code</th><th rowspan="2">Failure Scenarios</th><th colspan="2">CPC/LCP</th><th>SS/IPSC</th></tr>
<tr><th>New Call On Rest Channel</th><th>Callback during call hang (in call session)</th><th>New Call or Callback during call hang</th></tr>
<tr><td>0x0C</td><td>Transmit Interrupted Call Failure</td><td>The WL Call Setup request w/ interrupt access is failed to interrupt the ongoing OTA interrupt voice call.</td><td colspan="3">The third party application may indicate to the user that the call request failed.
or
It may retry with access criteria to takeover the on-going call.</td></tr>
<tr><td>0x0D</td><td>Higher Priority Call Takeover Failure</td><td>The call being sent from the third party application is preempted by another call with higher priority such as Emergency call.</td><td colspan="3">The third party application indicates to the user that the call has ended.

The third party applications call is no longer being transmitted.</td></tr>
<tr><td>0x0E – 0x80</td><td>RESERVED</td><td></td><td colspan="3"></td></tr>
<tr><td>0x81</td><td>Local Group Call not allowed</td><td>The WL Call Setup request for starting a Local Group call is declined because the site where the request is sent is reserved for Wide Area or Private calls.</td><td>The third party application may indicate to the user that the call request was unsuccessful.</td><td>n/a</td><td>n/a</td></tr>
<tr><td>0x82</td><td>Non-Rest Channel Repeater</td><td>The WL Call Setup request is received on the non-rest channel repeater.</td><td>The third party application waits for 90ms then auto-retries.

The retry to access the new rest channel.</td><td>The third party application may indicate to the user that the call has ended.

The third party application request is received after the call hang has expired. The third party application needs to setup call</td><td>n/a</td></tr>
</table>

Value	Reason Code	Failure Scenarios	CPC/LCP		SS/IPSC
			New Call On Rest Channel	Callback during call hang (in call session)	New Call or Callback during call hang
				on the rest channel.	
0x83	Destination Site/Sites Busy	The WL Call Setup request to start a wide area group call is declined because the destination sites of the group do not have channels available	The third party application may indicate to the user that the call request was unsuccessful.	n/a	n/a.
0x84	The repeater end the call – due to under-run	The repeater, to which the third party application is sending the call, ends the call due to jitter buffer under-runs continuously for over 720ms. This may due to the network congestion.	The third party application indicates to the user that the call has ended. *The third party applications call is no longer being transmitted.*		
0x85	Undefined Call Failure	Any other failures.	The third party application waits for 90ms then auto-retries. *The retry to access the new rest channel.*	The third party application auto-retries.	The third party application auto-retries.
0X86 – 0xA5	RESERVED				
0xA6	All call is ongoing or	The WL Call Setup request	The third party application may	n/a	n/a

Value	Reason Code	Failure Scenarios	CPC/LCP		SS/IPSC
			New Call On Rest Channel	Callback during call hang (in call session)	New Call or Callback during call hang
	in-progress	is declined because an All Call is ongoing.	indicate to the user that the call request was unsuccessful.		
0xA7 – 0xFF	RESERVED				

Table 57: Recommendation to Third Party Application on Failure Reason

Note: All the messages at the Call Control interface are transmitted in UDP/IP packet. The third party application shall have logic in place to handle the possible packet lost case.

6.8.28 Manufacturer's ID (MFID)

This field conveys the DMR Feature ID / Manufacturer's ID.

Maufacturer's ID Value	Allocation
0x00	Standard Feature
0x01	Reserved
0x10	Motorola Solutions Proprietary Feature
0x11-FF	Reserved

Table 58: Manufacturer's ID Allocation

6.8.29 RTP Header Information Field

This field indicates the 0 - 11 Bytes of RTP Version 2 header. The CSRC field is not present.

0 1	2	3	4 7	8	9 15	16 31
Ver	P	X	CC	M	PT	Sequence Number
Timestamp						
SSRC						

Figure 83 – RTP Header information Format

Ver (Version): 2 bits. It is always set to 2.

P (Padding): 1 bit. When this bit is set, this packet contains one or more additional padding bytes at the end, which are not part of the payload. The last byte of the padding contains a count of how many padding bytes should be ignored. Padding may be needed by some encryption algorithms with fixed block sizes or for carrying several RTP packets in a lower-layer protocol data unit.

X (Extension): 1 bit. If it is set, the fixed header is followed by exactly one header extension.

CC (CSRC count): 4 bits. The number of CSRC identifiers that follow the fixed header. A CSRC is not used in the IP Site Connect system, so this count is set to zero.

M (Marker): 1 bit. The interpretation of the marker is defined by a profile. It is intended to allow significant events such as frame boundaries to be marked in the packet stream. A profile may define additional marker bits or specify that there is no marker bit by changing the number of bits in the payload type field. This Marker is not used in IP Site Connect system. The third party application should ignore the setting of this Marker.

PT (payload type): 7 bits. MOTOTRBO repeater defines two specific payload types for IP Site Connect RTP payload.

Payload Type Value	Description
0x5D	Indicates an ongoing voice call. Apply to all payloads except the last RTP frame.
0x5E	Indicates the RTP payload is the last RTP payload frame or a single CSBK.

Table 59 – Payload Type definition

Sequence Number: 16 bits. The sequence number increments by one for each RTP data packet sent, and may be used by the receiver to detect packet loss and to restore packet sequence. The initial value of the sequence number is random (unpredictable) to make known-plaintext attacks on encryption more difficult, even if the source itself does not encrypt, because the packets may flow through a translator that does.

Timestamp: 32 bits. It is incremented by 480 (60ms * 8000 kHz).The timestamp reflects the sampling instant of the first octet in the RTP data packet. The sampling instant must be derived from a clock that increments monotonically and linearly in time to allow synchronization and jitter calculations. Note that MOTOTRBO repeater does not maintain a real time clock, so the timestamp is a relative time for a specific repeater. And this relative time is not synchronized with any network protocol such as Internet Time Protocol because the repeater may not have access time servers in a private network.

SSRC (Synchronization source): 32 bits. This field is not used system and should be set to 0.

6.8.30 Burst Type

This field indicates the DMR voice burst type.

Value	Allocation
0x00	Reserved
0x01	Voice Burst A
0x02	Voice Burst B
0x03	Voice Burst C
0x04	Voice Burst D
0x05	Voice Burst E
0x06	Voice Burst F
0x07	Voice Terminator
0x08	Privacy Header
0x09-0xFF	RESERVED

Table 60: DMR voice burst type Allocation

6.8.31 Service Option

This field indicates the Service Option information.

bit	Allocation
7	0:Non-emergency service 1:Emergency service
4-6	Reserved
3	Only for group call 0: Non-broadcast service 1: Broadcast service
0-2	Reserved

Table 61: Service Option Allocation

6.8.32 AMBE Encoded Voice Frames

This field indicates AMBE Encoded Voice Frames. A DMR voice burst contains three 20ms vocoder compressed frames. Every frame will contains 49 bits AMBE encoded data.

R: RESERVED

Bad Voice Frame:

0: AMBE decoder corrected all the bit errors in that 20 ms voice frame.

1: AMBE decoder could not correct all the bit errors in that 20 ms voice frame.

AMBE frame 1 bit 0 – bit 49: v1u_0 (0) – v1u_3 (12)

AMBE frame 2 bit 0 – bit 49: v2u_0 (0) – v2u_3 (12)

AMBE frame 3 bit 0 – bit 49: v3u_0 (0) – v3u_3 (12)

Offset	Allocation							
	MSB 7	6	5	4	3	2	1	0 LSB
0	R	R	R	R	R	R	R	Bad Voice Frame
1	v1u_0(11) v1u_0(10) v1u_0(9) v1u_0(8) v1u_0(7) v1u_0(6) v1u_0(5) v1u_0(4)							
2	v1u_0(3) v1u_0(2) v1u_0(1) v1u_0(0) v1u_1(11) v1u_1(10) v1u_1(9) v1u_1(8)							
3	v1u_1(7) v1u_1(6) v1u_1(5) v1u_1(4) v1u_1(3) v1u_1(2) v1u_1(1) v1u_1(0)							
4	v1u_2(11) v1u_2(10) v1u_2(9) v1u_2(8) v1u_2(7) v1u_2(6) v1u_2(5) v1u_2(4)							
5	v1u_2(3) v1u_2(2) v1u_2(1) v1u_2(0) v1u_3(12) v1u_3(11) v1u_3(10) v1u_3(9)							
6	v1u_3(8) v1u_3(7) v1u_3(6) v1u_3(5) v1u_3(4) v1u_3(3) v1u_3(2) v1u_3(1)							
7	v1u_3(0)	Bad Voice Frame	v2u_0(11) v2u_0(10) v2u_0(9) v2u_0(8) v2u_0(7), v2u_0(6)					
8	v2u_0(5) v2u_0(4) v2u_0(3) v2u_0(2) v2u_0(1) v2u_0(0) v2u_1(11) v2u_1(10)							
9	v2u_1(9) v2u_1(8) v2u_1(7) v2u_1(6) v2u_1(5) v2u_1(4) v2u_1(3) v2u_1(2)							
10	v2u_1(1) v2u_1(0) v2u_2(11) v2u_2(10) v2u_2(9) v2u_2(8) v2u_2(7) v2u_2(6)							
11	v2u_2(5) v2u_2(4) v2u_2(3) v2u_2(2) v2u_2(1) v2u_2(0) v2u_3(12) v2u_3(11)							
12	v2u_3(10) v2u_3(9) v2u_3(8) v2u_3(7) v2u_3(6) v2u_3(5) v2u_3(4) v2u_3(3)							
13	v2u_3(2) v2u_3(1) v2u_3(0)			Bad Voice Frame	v3u_0(11) v3u_0(10) v3u_0(9) v3u_0(8)			
14	v3u_0(7) v3u_0(6) v3u_0(5) v3u_0(4) v3u_0(3) v3u_0(2) v3u_0(1) v3u_0(0)							
15	v3u_1(11) v3u_1(10) v3u_1(9) v3u_1(8) v3u_1(7) v3u_1(6) v3u_1(5) v3u_1(4)							
16	v3u_1(3) v3u_1(2) v3u_1(1) v3u_1(0) v3u_2(11) v3u_2(10) v3u_2(9) v3u_2(8)							
17	v3u_2(7) v3u_2(6) v3u_2(5) v3u_2(4) v3u_2(3) v3u_2(2) v3u_2(1) v3u_2(0)							
18	v3u_3(12) v3u_3(11) v3u_3(10) v3u_3(9) v3u_3(8) v3u_3(7) v3u_3(6) v3u_3(5)							
19	v3u_3(4) v3u_3(3) v3u_3(2) v3u_3(1) v3u_3(0)						R (bits 2-0)	

Table 62: AMBE Encoded Voice Frames Allocation

6.8.33 Raw RSSI Information

This field indicates the Raw RSSI Value, which is the repeater internal Digital Signaling Porcessing reading in dBm without including the front end gain and step attenuation. To get the RSSI value at repeater receiver port, which can be retrieved by the XCMP command of Radio Status at the repeater XCMP interface (See Reference [3] for more information), the following conversion shall be applied:

RSSI(dBm) = DSPR (dbm) + STEPAttenuationValue (dB) - dBFS2dBm – FEGain (dB)

Where
RSSI = actual RF level at repeater receiver port
DSPR = DSP reading from Abacus I and Q = 10* log(I^2 + Q^2), which is what you get from the Call Control Packet
dBFS2dBm = Conversion constant = 44 (DVGA & Analog VGA engaged)
FEGain = Measured front End Gain Ahead of Abacus less Abacus attenuation.

Step Attenuation value, FEGain and dBFS2dBm are constant values. Each repeater might have different values. Currently we do not have an interface to allow the third party application to retrieve those values from the repeater. One workaround can be the third party application reads the RSSI value from the XCMP interface while receiving the Call Control packet, and derives those unknown constant values by subtracting the DSPR from the RSSI(dBm) in the above formula.

Keep in mind, the RSSI(dBm) in the above formula is the RF level at the repeater receiver port, which includes the external antennae gain. Some measurements in the field are needed to get the external antennae gain for each repeater setup. The third party application can subtract the external antennae gain from the RSSI(dBm) at the repeater receiver port to get the actual RF level at the external antennae, which is nice to have but not required.

6.8.34 Algorithm ID, Key ID and IV

Those fields indicate the Algorithm ID, Key ID and Initialization Vector use for enhanced privacy voice call.

Value	Allocation
0x00	Invalid algorithm ID when CRC error is detected.
0x01	ARC4
0x02-0xFF	RESERVED

Table 63 : Algorithm ID Allocation

Value	Allocation
0x00	Invalid Key ID when CRC error is detected.
0x01-0xFF	Valid ranges.

Table 64 : Key ID Allocation

Value	Allocation
0x00	Invalid IV when CRC error is detected.
0x01-0xFFFFFFFF	Valid ranges.

Table 65 : Initialization Vector Allocation

Below table shows how to set the Algorithm ID, Key ID and Initialization Vector in WL inbound/outbound call control PDU.

PDU	Inbound	Outbound
WL_VC_PRIVACY_BURST	AlgorithmID, keyID and IV used for encrypting the this voice super frame	AlgorithmID, keyID and IV used for decrypting the this voice super frame
WL_VC_VOICE_BURST Burst A,B,C,D and E	AlgorithmID = 0 key ID = 0 IV = 0	AlgorithmID = 0 key ID = 0 IV = 0
WL_VC_VOICE_BURST Burst F	AlgorithmID, keyID and IV used for encrypting the next voice super frame.	AlgorithmID, keyID and IV used for decrypting the next voice super frame.

Table 66 : Algorithm ID, Key ID and Initialization Vector Setting.

7 Motorola Solutions Proprietary Message Definitions

7.1 DMR CSBK Messages

7.1.1 Single Block Packet Description

The CSBK message contains a 96-bit information field. The general structure of the CSBK message is shown in Figure below.

Bit	7	6	5	4	3	2	1	0
Octet 0	LB	PF	Opcode					
Octet 1	Manufacturer's ID							
Octet 2 – Octet 9	Arguments							
Octet 10 – Octet 11	CRC							

CSBK message structure

- Last block flag (LB):

This flag indicates whether more CSBKs should be expected in this packet.
0: other CSBKs to follow for this packet
1: last (only) CSBK for this packet
It is always set to 1 for MOTOTRBO.

- Protected flag (PF):

This flag designates the protection mode for this CSBK.
0: non-protected mode
1: protected mode
It is always set to 0 for MOTOTRBO.

- Opcode:

It specifies the type of the message. The Data field is entirely dependent on the opcode. Motorola Solutions proprietary CSBK messages are shown as below.

Opcode	Value
ACK_RSP_U	%100000
CALL_ALERT_REQ	%011111
EMRG_ALRM_REQ	%100111
RAD_MON_COM	%011101
EXT_FNCT_CMD	%100100
EXT_FNCT_RSP	%100100

Table 67 – CSBK Opcode Definitions

For ETSI DMR standard CSBK messages:

- Preamble CSBK
- Negative Acknowledgement Response (NACK_RSP_U)
- Unit to Unit Voice Service Request (UU_V_REQ)
- Unit to Unit Voice Service Answer Response (UU_ANS_RSP)

Please refer to Reference [4] and [5] for more information.

- Manufacturer's ID (MFID):

It identifies the manufacturer for non-standard control channel messaging. For the Motorola Solutions proprietary CSBK, the MFID is 0x10. And for the ETSI DMR standard CSBK, the MFID is 0x00.

MFID Value	Allocation
0x00	Standard
0x10	Motorola Solutions Proprietary

For Motorola Solutions proprietary CSBK messages, their message definition is described in the following sections.

- CRC

This is the CRC parity check. It provides error detection for the information of this CSBK (Octets 0-9). For the CRC calculation, please refer to Reference [3], section B.3.8, for more information.

7.1.2 Acknowledge Response - Unit (ACK_RSP_U)

This is the generic response to acknowledge an action when there is no other expected response.

Bit	7	6	5	4	3	2	1	0
Octet 0	LB	PF	Opcode					
Octet 1	Manufacturer's ID							
Octet 2	0	0	Service Type					
Octet 3	Reserved							
Octet 4	Target Address							
Octet 5								
Octet 6								
Octet 7	Source Address							
Octet 8								
Octet 9								
Octet 10	CRC							
Octet 11								

Figure 84 - Acknowledge Response - Unit

7.1.3 Call Alert Request (CALL_ALERT_REQ)

This message is used to command a radio to execute a Call Alert request operation.

Bit	7	6	5	4	3	2	1	0
Octet 0	LB	PF	Opcode					
Octet 1	Manufacturer's ID							
Octet 2	Reserved							
Octet 3								
Octet 4	Target Address							
Octet 5								
Octet 6								
Octet 7	Source Address							
Octet 8								
Octet 9								
Octet 10	CRC							
Octet 11								

Figure 85 – Call Alert Request

7.1.4 Emergency Alarm Request (EMRG_ALRM_REQ)

This is a special status indication typically reserved for the "life threatening" situation.

Bit	7	6	5	4	3	2	1	0
Octet 0	LB	PF	Opcode					
Octet 1	Manufacturer’s ID							
Octet 2	Reserved				EASN			
Octet 3	Reserved							
Octet 4	Group Address							
Octet 5								
Octet 6								
Octet 7	Source Address							
Octet 8								
Octet 9								
Octet 10	CRC							
Octet 11								

Figure 86 – Emergency Alarm Request

7.1.5 Radio Remote Monitor command (RAD_MON_CMD):

This message is used to command a radio to execute a Radio Unit Remote Monitor operation.

Bit	7	6	5	4	3	2	1	0
Octet 0	LB	PF	Opcode					
Octet 1	Manufacturer’s ID							
Octet 2	Reserved							
Octet 3	Reserved						TX Mult.	
Octet 4	Source Address							
Octet 5								
Octet 6								
Octet 7	Target Address							
Octet 8								
Octet 9								
Octet 10	CRC							
Octet 11								

Figure 87 – Radio Remote Monitor Command

The TX Multiplier is a 2-bit value ranging from 0 to 3. It multiplies a stored value programmed in the target radio to represent the requested time to key the transmitter during the monitor function. The zero value does not cause the radio to key.

7.1.6 Extended Function Command (EXT_FNCT_CMD)

This is the transaction addressed to an SU for an extended function.

Bit	7	6	5	4	3	2	1	0
Octet 0	LB	PF	Opcode					
Octet 1	Manufacturer’s ID							
Octet 2 – Octet 6	Extended Function							
Octet 7 – Octet 9	Target Address							
Octet 10 – Octet 11	CRC							

Figure 88 – Extended Function Command

7.1.7 Extended Function Response (EXT_FNCT_RSP)

This transaction is the response to an Extended Function command.

Bit	7	6	5	4	3	2	1	0
Octet 0	LB	PF	Opcode					
Octet 1	Manufacturer’s ID							
Octet 2	Extended Function							
Octet 3								
Octet 4								
Octet 5								
Octet 6								
Octet 7	Source Address							
Octet 8								
Octet 9								
Octet 10	CRC							
Octet 11								

Figure 89 – Extended Function Response

7.1.8 Field Definitions

7.1.8.1 Service Type

The Service Type field indicates the service which is being identified. This is set equal to the appropriate CSBK Opcode value (defined in Table 67) for the identified service.

7.1.8.2 Emergency Alarm Sequence Number (EASN)

The Emergency Alarm Sequence Number field is a 4-bit value ranges from 0 to 15. This field shall only increment when an emergency is first initiated. Multiple attempts of the same emergency alarm message shall have the same Emergency Alarm Sequence Number. Default value shall be 0.

7.1.8.3 Target Address

This field identifies the individual subscriber unit which is the destination of the CSBK. This is a 24-bit vector which uniquely identifies the subscriber unit within the System. It shall utilize the Subscriber Unit address definitions.

7.1.8.4 Source Address

This field identifies the individual subscriber unit which originates the CSBK. This is a 24-bit vector which uniquely identifies the subscriber unit within the System. It shall utilize the Subscriber Unit address definitions.

7.1.8.5 Extended Function

The Extended Function is a collection of related functions and operations. The Extended Function field is composed of the following subfields:

Class (1 byte)
Operand (1 byte)
Arguments (3 bytes)

Class will determine the type of extended function to be considered.

Operand will determine the actual function being addressed based upon the Class designation.

Arguments will supply additional processing information. This may not be required for all extended functions, and will be set to "null" (0) if not required.

Class	Operand	Arguments	Description
0x00	0x00	Source Address	Radio Check
0x00	0x01 – 0x7C	Reserved	Reserved
0x00	0x7D	Source Address	Reserved
0x00	0x7E	Source Address	Radio Uninhibit
0x00	0x7F	Source Address	Radio Inhibit
0x00	0x80	Target Address	Radio Check ACK
0x00	0x81	Source Address	IP Console Radio Un-inhibit
0x00	0x82	Source Address	IP Console Radio Inhibit
0x00	0x83	Target Address	IP Console Radio Un-inhibit Acknowledgement
0x00	0x84	Target Address	IP Console Radio Inhibit Acknowledgement
0x00	0x85 – 0xFC	Reserved	Reserved
0x00	0xFD	Source Address	Reserved
0x00	0xFE	Target Address	Radio Uninhibit ACK
0x00	0xFF	Target Address	Radio Inhibit ACK

Table 68 – Extended Function Values

Appendix A: LFSR Sample Implementation

For a 32 bit LFSR, the implementation is as follows: x^32 + x^4 + x^2 + 1.

```
/*--------------------------- DESCRIPTION OF LOGIC -------------------------
*
* 4 8-bit shifts
*          For N 8-bit shifts use polynomial formula x^32 + x^4 + x^2 + 1
*       Since there are two taps in bit 3 and bit 1, we need to
*       operative two bytes at a time. temp1, temp3, temp5, temp 7
*       temp1 stores bit 7, bit 6, temp3 stores bit 5, bit 4,
*       temp5 stores bit 3, bit 2, temp7 stores bit 1, bit 0
*       temp 1 = Calculate bit 31/bit 30 ^ bit 3/bit 2 ^ bit 1/bit 0;
*       temp 3 = Calculate bit 29/bit 28 ^ bit 1/bit 0 ^ temp1>>6
*       temp 5 = Calculate bit 27/bit 26 ^ temp 3>> 6 ^ temp2 << 2
*       temp 7 = Calculate bit 25/bit 24 ^ temp 5>>2 ^ temp4 << 2
*       Shift bits 23 - 16 to bits 31 - 24
*       Shift bits 15 - 8 to bits 23 - 16
*       Shift bits 7 - 0 to bits 15 - 8
*       Store temp1 | temp3 | temp 5 | temp 7 in bits 7 - 0
*   EndFor
***************************************************************************/

unsigned char i;
unsigned char temp, temp1, temp2, temp3, temp4, temp5, temp6, temp7;

//4 byte IV length
{
        for (i = 0; i < shifts; i++)
        {
                temp1 = ( (dataPtr[0]) ^  ((dataPtr[3] << 4)) ^ ((dataPtr[3] << 6)) ) & 0xC0;

                temp2 = (temp1 >> 6) | (dataPtr[3] << 2);
                temp3 = ( (dataPtr[0] ) ^ ((temp2 << 2)) ^ ((temp2 << 4))) & 0x30;
                temp4 = (temp3 >> 4) | (temp2 << 2);
                temp5 = ((dataPtr[0] ) ^ ((temp4)) ^ ((temp4 << 2))) & 0x0C;

                temp6 = (temp5 >> 2) | (temp4 << 2);
                temp7 = ((dataPtr[0]) ^ ((temp6)) ^ ((temp6 >> 2))) & 0x03;

                dataPtr[0] = dataPtr[1];
                dataPtr[1] = dataPtr[2];

                dataPtr[2] = dataPtr[3];

                dataPtr[3] = temp1 | temp3 | temp5 | temp7;
        }
}
```

Appendix B: Golay(24, 12, 8) Matrix

```
{ 1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 0, 1 },
{ 0, 1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 1 },
{ 1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 0, 0 },
{ 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 0 },
{ 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0 },
{ 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 1 },
{ 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 1 },
{ 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 1 },
{ 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0 },
{ 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 1, 1 },
{ 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 1, 0 },
{ 1, 0, 0, 0, 1, 1, 1, 0, 1, 0, 1, 1 }
```

3137 # Appendix C: