Research & Vehicle Technology
"Infotainment Systems Product Development"

Feature – Alerts v3 (Chimes/Prompts/Beeps/Audio Attenuation)

APIM Infotainment Subsystem Part Specific Specification (SPSS)

Version 1.1

UNCONTROLLED COPY IF PRINTED

Version Date: March 17, 2020

FORD CONFIDENTIAL

Revision History

Date	Version	Notes	
December 4, 2019	**1.0**	**Initial SYNC 4.2 Release**	**Integrates AHU and Cluster features into SYNC**
March 17, 2020	**1.1**		
	STR-681029/B-Chime Activation Requirements		<jmyslin2> removed Chime_Not_Recognized variant 1 requirement since SYNC 4.2 only supports variant 2
	ALERTv2-SR-REQ-376339/A-Infotainment System Chime Source - Variant 2		jmyslin2: New requirement
	ALERT-SR-REQ-368145/B-_Chime_Not_Recognized signal - variant 2		jmyslin2: added note to requirement
	ALERTv2-SR-REQ-361649/B-Loss of communication with Chime module - variant 2		jmyslin2: updated requirement
	ALERTv2-SR-REQ-361651/B-Chime Error States/ Fault handling - variant 2		<jmyslin2> updated requirement
	ALERTv2-SR-REQ-372147/A-_Chime_Supported signal changes to Supported from Not_Supported - variant 2		<jmyslin2> New requirement
	ALERTv2-SR-REQ-372081/A-Load Shed - SYNC 4.2		<jmyslin2> new requirement
	PWRMAN-SR-REQ-014509/F-Infotainment Components Load Shed State requirements (TcSE ROIN-66172-3)		jmyslin2: added clarification to the requirement for SYNC 4.2
	ALERT-SR-REQ-376198/A-Transport Mode - SYNC 4.2		<jmyslin2> new requirement

Table of Contents

REVISION HISTORY **2**

1 ARCHITECTURAL DESIGN **5**

1.1 Interface Requirements - Alert APIM variant 3 ..... *5*
1.1.1 MD-REQ-275630/B-IPC_Infotainment 5
1.1.2 MD-REQ-275638/B-IPC_Chime 5
1.1.3 MD-REQ-275692/B-IPC_Chime2 7
1.1.4 MD-REQ-275693/B-AHU_Alert 9
1.1.5 MD-REQ-275701/B-DSP_AMP_Alert 10
1.1.6 MD-REQ-275704/B-SYNC_Alerts 11
1.1.7 ALERT-MD-REQ-232947/E-rearParkAidChimeVolumeSelection - FBMP 11
1.1.8 ALERT-MD-REQ-233050/C-frontParkAidChimeVolumeSelection - FBMP 12

1.2 ALERT-CLD-REQ-014717/C-Chime Client (TcSE ROIN-202535-1) *13*

1.3 ALERT-CLD-REQ-014718/B-Chime Generator Server (TcSE ROIN-202536-1) *13*

1.4 ALERT-CLD-REQ-014719/B-Chime Audio Source Server (TcSE ROIN-202537-1) *13*

1.5 ALERT-CLD-REQ-014720/B-Prompt Client/Generator Server (TcSE ROIN-202538-1) *13*

1.6 ALERT-CLD-REQ-014721/B-Prompt Audio Source Server (TcSE ROIN-202539-1) *13*

1.7 ALERT-CLD-REQ-014722/B-Beep Client (TcSE ROIN-202534-1) *13*

1.8 ALERT-CLD-REQ-014723/B-Beep Generator Server (TcSE ROIN-202533-1) *13*

1.9 ALERT-CLD-REQ-014724/B-Beep Audio Source Server (TcSE ROIN-202532-1) *13*

1.10 ALERT-CLD-REQ-232214/B-Adjustable Chime Client *13*

1.11 ALERT-CLD-REQ-232215/B-Adjustable Chime Server *13*

2 GENERAL REQUIREMENTS **14**

2.1 ALERTv2-SR-REQ-095179/G-Alert Configuration Table *14*

3 FUNCTIONAL DEFINITION **16**

3.1 ALERTv2-FUN-REQ-361260/A-Chimes - variant 2 (with integrated Chime Client, Generator & Audio Source) *16*
3.1.1 Overview 16
3.1.2 Deployment chimes variant 2 (for reference only) 16
3.1.3 Chime Activation Requirements 17
3.1.4 Chimes Error Management Requirements 27
3.1.5 Sequence Diagrams 30

3.2 ALERT-FUN-REQ-195348/C-Adjustable Chime Strategy *41*
3.2.1 Use Cases 41
3.2.2 Requirements 45
3.2.3 Sequence Diagrams 49

3.3 ALERT-FUN-REQ-348785/A-Mixable Prompts - Variant 2 ..... *52*
3.3.1 Requirements 52
3.3.2 Sequence Diagrams 53

3.4 ALERTv2-FUN-REQ-095181/B-Beep *55*
3.4.1 Beep Activation Requirements 55
3.4.2 Sequence Diagrams 57

3.5 ALERT-FUN-REQ-014794/C-Audio Attenuation/Muting Strategy (TcSE ROIN-119789-1) *58*
3.5.1 Cluster Prioritizing Audio Attenuation Requests 58
3.5.2 AHU / DSP AMP Prioritizing Audio Attenuation Requests 58
3.5.3 Infotainment System Audio Attenuation 58

The information contained in this document is Proprietary to Ford Motor Company.

4 APPENDIX: REFERENCE DOCUMENTS 62

FORD MOTOR COMPANY CONFIDENTIAL
The information contained in this document is Proprietary to Ford Motor Company.

1 Architectural Design

1.1 Interface Requirements - Alert APIM variant 3

1.1.1 MD-REQ-275630/B-IPC_Infotainment

Message Type: Status

Method sent from the Chime Client to the Infotainment modules for Chime Power Moding, Chime Audio Source and Chime Audio Attenuation

Logical Signal Name	Literals	Value	Description
Chime_Source	Inactive/Invalid	0x0	Chime audio source signal
	Infotainment_System	0x1	
	Cluster	0x2	
Power_Up_Chime_Modules	Inactive	0x0	Chime power moding signal
	Active	0x1	
Attn_Info_Audio	No Attenuation of Audio	0x0	Audio Attenuation signal
	Attenuation_1	0x1	
	Attenuation_2	0x2	
	cont.		
	Attenuation_6	0x6	
	Unknown	0x7	
New_Attn_Event	Inactive	0x0	New attenuation event signal
	Active	0x1	

1.1.2 MD-REQ-275638/B-IPC_Chime

Message Type: Request

Request from the Chime Client to the Infotainment chime modules (Chime Generator Server and Chime Audio Source Server) to play a particular chime

Logical Signal Name	Literals	Value	Description
Chime	Inactive	0x0	Chime Sound ID
	No_Chime	0x1	
	Chime_1	0x2	
	Chime_2	0x3	
	Chime_3	0x4	
	cont.		
	Chime_62	0x3F	
Chime_Vol_Level	100% of stored chime volume / Inactive	0x0	Chime Volume Level
	99% of stored chime volume	0x1	
	98% of stored chime volume	0x2	
	97% of stored chime volume	0x3	
	cont.		
	2% of stored chime volume	0x62	
	1% of stored chime volume	0x63	
	Reserved	0x64	
Chime_Directionality	OFF / Inactive	0x0	Chime Speaker Directionality

	All	0x1	
	Front	0x2	
	Rear	0x3	
Chime_Time_Criticality	Inactive	0x0	Chime Criticality
	Criticality_High	0x1	
	Criticality_Low	0x2	
OFF_Time_Btwn_Chime	Inactive	0x0	Off Time Between Chimes
	Continuously (repeat continuously)	0x1	
	5 msec	0x2	
	10 msec	0x3	
	15 msec	0x4	
	20 msec	0x5	
	40 msec	0x6	
	60 msec	0x7	
	conti… increment in 20 msec until 800 msec		
	800 msec	0x2C	
	850 msec	0x2D	
	900 msec	0x2E	
	1 sec	0x2F	
	1.5 sec	0x30	
	2.0 sec	0x31	
	3 sec	0x32	
	4 sec	0x33	
	5 sec	0x34	
Chime_Occurence	Inactive	0x0	Determines the number of repetitions the Chime_X sound to be played, or for the Chime_X to be played continuously
	Play Once	0x1	
	2 Repetitions	0x2	
	3 Repetitions	0x3	
	4 Repetitions	0x4	
	5 repetitions	0x5	
	cont…		
	14 repetitions	0xE	
	Continuous	0xF	
AdjustableChimeVol	Default Volume / Inactive / 0 adjustable chime volume level	0x0	Adjustable Chime Volume Level
	-6 Reserved, do not use	0x1	
	-5 Reserved, do not use	0x2	
	-4 Reserved, do not use	0x3	
	-3 adjustable chime volume level	0x4	
	-2 adjustable chime volume level	0x5	
	-1 adjustable chime volume level	0x6	
	+1 adjustable chime volume level	0x7	

	+2 adjustable chime volume level	0x8	
	+3 adjustable chime volume level	0x9	
	+4 Reserved, do not use	0xA	
	+5 Reserved, do not use	0xB	
	+6 Reserved, do not use	0xC	
	Reserved, do not use	0xD	
	Reserved, do not use	0xE	
	Reserved, do not use	0xF	
ChimeID_No_Rq	Inactive	0x0	Chime ID
	ID 1	0x1	
	ID 2	0x2	
	ID 3	0x3	
	cont.		
	ID 255	0xFF	

1.1.3 MD-REQ-275692/B-IPC_Chime2

Message Type: Request

Request from the Chime Client to the Infotainment chime modules (Chime Generator Server and Chime Audio Source Server) to play a particular chime

Logical Signal Name	Literals	Value	Description
Chime2	Inactive	0x0	Chime Sound ID
	No_Chime	0x1	
	Chime_1	0x2	
	Chime_2	0x3	
	Chime_3	0x4	
	cont.		
	Chime_62	0x3F	
Chime_Vol_Level2	100% of stored chime volume / Inactive	0x0	Chime Volume Level
	99% of stored chime volume	0x1	
	98% of stored chime volume	0x2	
	97% of stored chime volume	0x3	
	cont.		
	2% of stored chime volume	0x62	
	1% of stored chime volume	0x63	
	Reserved	0x64	
Chime_Directionality2	OFF / Inactive	0x0	Chime Speaker Directionality
	All	0x1	
	Front	0x2	
	Rear	0x3	
Chime_Time_Criticality2	Inactive	0x0	Chime Criticality
	Criticality_High	0x1	
	Criticality_Low	0x2	
	Inactive	0x0	Off Time Between Chimes

OFF_Time_Btwn_Chime 2	Continuously (repeat continuously)	0x1	
	5 msec	0x2	
	10 msec	0x3	
	15 msec	0x4	
	20 msec	0x5	
	40 msec	0x6	
	60 msec	0x7	
	conti… increment in 20 msec until 800 msec		
	800 msec	0x2C	
	850 msec	0x2D	
	900 msec	0x2E	
	1 sec	0x2F	
	1.5 sec	0x30	
	2.0 sec	0x31	
	3 sec	0x32	
	4 sec	0x33	
	5 sec	0x34	
Chime_Occurence2	Inactive	0x0	Determines the number of repetitions the Chime2_X sound to be played, or for the Chime2_X to be played continuously
	Play Once	0x1	
	2 Repetitions	0x2	
	3 Repetitions	0x3	
	4 Repetitions	0x4	
	5 repetitions	0x5	
	cont…		
	14 repetitions	0xE	
	Continuous	0xF	
AdjustableChimeVol2	Default Volume / Inactive / 0 adjustable chime volume level	0x0	Adjustable Chime Volume Level
	-6 Reserved, do not use	0x1	
	-5 Reserved, do not use	0x2	
	-4 Reserved, do not use	0x3	
	-3 adjustable chime volume level	0x4	
	-2 adjustable chime volume level	0x5	
	-1 adjustable chime volume level	0x6	
	+1 adjustable chime volume level	0x7	
	+2 adjustable chime volume level	0x8	
	+3 adjustable chime volume level	0x9	
	+4 Reserved, do not use	0xA	
	+5 Reserved, do not use	0xB	
	+6 Reserved, do not use	0xC	
	Reserved, do not use	0xD	

	Reserved, do not use	0xE	
	Reserved, do not use	0xF	
Chime2ID_No_Rq	Inactive	0x0	Chime ID
	ID 1	0x1	
	ID 2	0x2	
	ID 3	0x3	
	cont.		
	ID 255	0xFF	

1.1.4 MD-REQ-275693/B-AHU_Alert

Message Type: Status

Alert Status from the Audio Head Unit to the applicable Alert components

Logical Signal Name	Literals	Value	Description
AHU_Chime_Supported	Inactive	0x0	Signal from the AHU indicating to the Chime Master (ie Cluster) if chimes are supported by the AHU
	Supported	0x1	
	Not_Supported	0x2	
AHU_Chime_Not_Recognized	Inactive	0x0	Signal from the AHU Chime Generator to Cluster letting the Cluster know the chime requested is not recognized. If the Chime is recognized the signal remains set to inactive.
	Chime_Not_Recognized	0x1	
AHU_Chime2_Not_Recognized	Inactive	0x0	
	Chime_Not_Recognized	0x1	
AHU_Chime_Active	Inactive	0x0	Signal from the AHU Chime Generator to the Cluster Chime Client indicating what chime is active. Note: the AHU sets this signal to No_Chime when it is not the Chime Generator (ex. when Cluster is Chime Generator because AHU_Chime_Supported = Not_Supported).
	No_Chime	0x1	
	Chime_1	0x2	
	Chime_2	0x3	
	Chime_3	0x4	
	cont.		
	Chime_62	0x3F	
AHU_Chime2_Active	Inactive	0x0	
	No_Chime	0x1	
	Chime_1	0x2	
	Chime_2	0x3	
	Chime_3	0x4	
	cont.		
	Chime_62	0x3F	
Chime_Alert_Chan	Inactive	0x0	Signal to mute the Chime Audio Source Alert channel after the Chime Generator is done producing the chime
	Mute	0x1	
Chime_Alert_Chan2	Inactive	0x0	
	Mute	0x1	
AHU_AdjustableChimeVolSupported	Inactive	0x0	Signal to indicate if a requested adjustable Chime value (ex +1, -2) for a particular Chime_X sound is supported or not
	Adjustable Chime Sound Not Supported	0x1	
	Adjustable Chime Sound Supported	0x2	
AHU_AdjustableChimeVolSupported2	Inactive	0x0	
	Adjustable Chime Sound Not Supported	0x1	

	Adjustable Chime Sound Supported	0x2	
ChimeID_No_Stat	Inactive	0x0	Signal for Chime ID
	ID1	0x1	
	ID2	0x2	
	ID3	0x3	
	cont.		
	ID255	0xFF	
Chime2ID_No_Stat	Inactive	0x0	
	ID1	0x1	
	ID2	0x2	
	ID3	0x3	
	cont.		
	ID255	0xFF	
Alert_Channel1	Inactive	0x0	Signal from the AHU Prompt Audio Source to the mixable Prompt Generator indicating the status of the Prompt Audio Source input Alert Channel. • Note set to Inactive if no separate Prompt Generator (ex no SYNC)
	Initialized for Prompts	0x1	
	Muted	0x2	
	Reserved	0x3	

1.1.5 MD-REQ-275701/B-DSP_AMP_Alert

Message Type: Status

Alert Status from the DSP AMP to the applicable Alert components

Logical Signal Name	Literals	Value	Description
DSP_Chime_Not_Recognized	Inactive	0x0	Signal indicating a Chime_X sound isn't recognized for a particular Chime request
	Chime_Not_Recognized	0x1	
DSP_Chime2_Not_Recognized	Inactive	0x0	
	Chime_Not_Recognized	0x1	
DSP_Chimes_Supported	Inactive	0x0	Signal indicating to the Chime Generator if chimes are supported by the DSP AMP
	Supported	0x1	
	Not_Supported	0x2	
Alert_Channel1	Inactive	0x0	Status for the Chime DSP Alert Channels
	Reserved	0x1	
	Muted	0x2	
	Initialized for Chime	0x3	
Alert_Channel2	Inactive	0x0	
	Reserved	0x1	
	Muted	0x2	
	Initialized for Chime	0x3	
DSP_AdjustableChimeVolSupported	Inactive	0x0	Signal to indicate if a requested adjustable Chime value (ex +1, -2) for a particular Chime_X sound is supported or not
	Adjustable Chime Sound Not Supported	0x1	
	Adjustable Chime Sound Supported	0x2	
DSP_AdjustableChimeVolSupported2	Inactive	0x0	

The information contained in this document is Proprietary to Ford Motor Company.

	Adjustable Chime Sound Not Supported	0x1	
	Adjustable Chime Sound Supported	0x2	
Alert_Channel3	Inactive	0x0	Status for the mixable prompt DSP AMP Alert Channel
	Initialized for Prompts	0x1	
	Muted	0x2	
	Reserved	0x3	

1.1.6 MD-REQ-275704/B-SYNC_Alerts

Message Type: Status

Signal from SYNC to the applicable Alert components

Logical Signal Name	Literals	Value	Description
Alert_Chan	OFF_Inactive (prompts off)	0x0	Signal from the SYNC Prompt Generator to the Prompt Audio Source to control the Alert channel for prompts
	Mute	0x1	
	Initialized for Prompts	0x2	
Prompt_Directionality	OFF / Inactive	0x0	Signal from the SYNC Prompt Generator to the Prompt audio source indicating what speaker(s) to play the prompt(s) through
	All	0x1	
	Front	0x2	
	Rear	0x3	
Attn_Info_Audio	No Attenuation of Audio	0x0	Signal from the SYNC Prompt Generator to the Prompt Audio Source for attenuating the active audio source.
	Attenuation_1	0x1	
	Attenuation_2	0x2	
	Attenuation_3	0x3	
	cont.		
	Attenuation_6	0x6	
	Unknown	0x7	
New_Attn_Event	Inactive	0x0	Signal From the SYNC Prompt Generator to the Prompt audio source for a new attenuation event
	Active	0x1	
Audible_Beeps	Inactive	0x0	Signal from the SYNC Beep Client to the Beep Generator so the Beep Generator can produce an audible beep
	Active	0x1	

1.1.7 ALERT-MD-REQ-232947/E-rearParkAidChimeVolumeSelection - FBMP

For this feature when performing the “Set” or “Query” operation the Feature Number and Configuration Number in the Feature.Rq and Feature.St messages shall be used below.

If Enhanced Memory is supported the Active Personality Profile shall be used for PersIndex. If Enhanced Memory is not supported PersIndex shall be set to Vehicle. See Feature Based Message Protocol SPSS for details.

Feature Description	Feature Number	Configuration Number	Configuration Name / HMI selection
		0x00	-6 Reserved / Do not use

The information contained in this document is Proprietary to Ford Motor Company.

Rear Park Aid Adjustable Volume Selection and Status rearParkAidChimeVolSelection signal in SPSS	0x0910	0x01	-5 Reserved / Do not use
		0x2	-4 Reserved / Do not use
		0x3	-3 adjustable volume level
		0x4	-2 adjustable volume level
		0x5	-1 adjustable volume level
		0x6	0 Adjustable chime volume / Default Volume
		0x7	+1 adjustable volume level
		0x8	+2 adjustable volume level
		0x9	+3 adjustable volume level
		0xA	+ 4 Reserved / Do not use
		0xB	+ 5 Reserved / Do not use
		0xC	+ 6 Reserved / Do not use

1.1.8 ALERT-MD-REQ-233050/C-frontParkAidChimeVolumeSelection - FBMP

For this feature when performing the “Set” or “Query” operation the Feature Number and Configuration Number in the Feature.Rq and Feature.St messages shall be used below.

If Enhanced Memory is supported the Active Personality Profile shall be used for PersIndex. If Enhanced Memory is not supported PersIndex shall be set to Vehicle. See Feature Based Message Protocol SPSS for details.

Feature Description	Feature Number	Configuration Number	Configuration Name / HMI selection
Front Park Aid Adjustable Volume Selection and Status frontParkAidChimeVolSelection signal in SPSS	0x0911	0x00	-6 Reserved / Do not use
		0x01	-5 Reserved / Do not use
		0x2	-4 Reserved / Do not use
		0x3	-3 adjustable volume level
		0x4	-2 adjustable volume level
		0x5	-1 adjustable volume level
		0x6	0 Adjustable chime volume / Default Volume
		0x7	+1 adjustable volume level
		0x8	+2 adjustable volume level

	0x9	+3 adjustable volume level
	0xA	+ 4 Reserved / Do not use
	0xB	+ 5 Reserved / Do not use
	0xC	+ 6 Reserved / Do not use

1.2 ALERT-CLD-REQ-014717/C-Chime Client (TcSE ROIN-202535-1)

The Chime Client is the chime master that tells the infotainment system what chime(s) to play

1.3 ALERT-CLD-REQ-014718/B-Chime Generator Server (TcSE ROIN-202536-1)

The Chime Generator Server produces the chime signal for a respective chime

1.4 ALERT-CLD-REQ-014719/B-Chime Audio Source Server (TcSE ROIN-202537-1)

The Chime Audio Source Server produces the chime sound heard in the infotainment system

1.5 ALERT-CLD-REQ-014720/B-Prompt Client/Generator Server (TcSE ROIN-202538-1)

The Prompt Client/Generator Server is responsible for the prompt function and will produce a prompt signal for the Prompt Audio Source

1.6 ALERT-CLD-REQ-014721/B-Prompt Audio Source Server (TcSE ROIN-202539-1)

The Prompt Audio Source Server is responsible for playing the prompt sound through the infotainment system

1.7 ALERT-CLD-REQ-014722/B-Beep Client (TcSE ROIN-202534-1)

The Beep Client requests the Beep Generator to produce the beep

1.8 ALERT-CLD-REQ-014723/B-Beep Generator Server (TcSE ROIN-202533-1)

The Beep Generate Server creates the signal for the beep

1.9 ALERT-CLD-REQ-014724/B-Beep Audio Source Server (TcSE ROIN-202532-1)

The Beep Audio Source Server produces the sound for the beep through the infotainment system

1.10 ALERT-CLD-REQ-232214/B-Adjustable Chime Client

The Adjustable Chime Client is the interface with the HMI and tells the Adjustable Chime Server what adjustable chime selections where made

1.11 ALERT-CLD-REQ-232215/B-Adjustable Chime Server

The Adjustable Chime Server receives the requests for adjustable chime levels from the Adjustable Chime Client to be used by the Chime Arbitrator/Chime Client to make the chime requests to the infotainment system.

2 General Requirements

2.1 ALERTv2-SR-REQ-095179/G-Alert_Configuration_Table

The Alert Generator and Alert Audio Source will vary depending on the infotainment system module availability. The Infotainment System Alert set-up will follow the table below.

Modules Present	Chime Client (Chime Master)	Beep Generator	Prompt Generator	Chime Generator	Alert Audio Source (Module to Play & Mix Audio)
IAHU / DSP AMP	Cluster	DSP AMP	IAHU	IAHU	DSP AMP
IAHU	Cluster	IAHU	IAHU	IAHU	IAHU

Table 1: Alert Configuration Table with separate Cluster and IAHU (DSP AMP optional)

~~Modules Present~~	~~Chime Client (Chime Master)~~	~~Beep Generator~~	~~Prompt Generator~~	~~Chime Generator~~	~~Alert Audio Source (Module to Play & Mix Audio)~~
~~ICI / DSP AMP~~	~~ICI~~	~~DSP AMP~~	~~ICI~~	~~ICI~~	~~DSP AMP~~
~~ICI~~	~~ICI~~	~~ICI~~	~~ICI~~	~~ICI~~	~~ICI~~

~~**Alert Configuration Table with Cluster and IAHU integrated**~~

	IAHU / DSP AMP variant 2	IAHU only	ICI / DSP AMP variant 2	ICI
Chime Client (Chime Master)	Cluster	Cluster	ICI	ICI
Chime Generator	IAHU	IAHU	ICI	ICI
Chime Audio Source	IAHU	IAHU	ICI	ICI
Adjustable Chime Client	IAHU	IAHU	ICI	ICI
Prompt Generator	IAHU	IAHU	ICI	ICI
Prompt Audio Source	DSP v2	IAHU	DSP v2	ICI
Beep Client	IAHU	IAHU	ICI	ICI
Beep Generator	IAHU	IAHU	ICI	ICI
Beep Audio Source	IAHU	IAHU	ICI	ICI
Audio Attenuation Server (ie Rx Attn_Info_Audio)	IAHU and DSP v2 (if necessary)	IAHU	ICI and DSP v2 (if necessary)	ICI

Table 2: Alert Configuration (DSP AMP variant 2 optional)

Note:

- IAHU stands for Integrated AHU. It references modules where the infotainment system master and AHU are integrated in one module (ex.Connected HMI Radio, SYNC integrated with AHU, Connected Touch Radio...).
- Alert Audio Source in table 1 includes beep audio source, prompt audio source and chime audio source.
- ICI stands for Integrated Cluster & IAHU (AHU and System Master)
- When this requirement was last written only table 2 applied to APIM 4.1 (IAHU) and APIM 4.2 (ICI)

- The DSP AMP and DSP AMP variant 2 are mutually exclusive.
- The DSP AMP variant 2 only needs to be the Audio Attenuation Server if the DSP AMP variant 2 generates any audio itself that needs to be muted/attenuated. The DSP AMP variant 2 is not required to use the Attn_Info_Audio signal to mute/attenuate audio sent to it from the IAHU (ex APIM 4.1/4.2). For the Attn_Info_Audio signal the IAHU would mute/attenuate the audio it sends to the DSP AMP variant 2.

FORD MOTOR COMPANY CONFIDENTIAL
The information contained in this document is Proprietary to Ford Motor Company.

3 Functional Definition

3.1 ALERTv2-FUN-REQ-361260/A-Chimes - variant 2 (with integrated Chime Client, Generator & Audio Source)

3.1.1 Overview

For Chimes variant 2 the feature integrates the Cluster and AHU functionality in one SYNC 4.2 ICI module and they are no longer separate modules on the CAN bus. The Cluster and AHU functionality are separate core processors in SYNC 4.2 at the time this was written so an internal interface must still be developed. From this spec the logical signals could be used to develop signals internal to SYNC 4.2 (ex logical IPC_chime message between Chime Client (VMCU) and Chime audio source (CCPU)).

The DSP AMP variant 2 is no longer a part of chime command and control (ie doesn't look at IPC_Chime/IPC_Chime2 message). Chime audio is simply passed from the SYNC 4.2 module to the DSP AMP variant 2. The DSP AMP variant 2 has audio routed to the speakers connected to it so chime diagnostics must still be performed.

Some additional requirements and updates were made from variant 1 of chimes (ie Alert-FUN-014731-Chimes) that are specific to chimes variant 2 called out in this spec.

Some chime arbitrator spec updates are needed that are not requirements in this spec but are still changes from chimes variant 1 (ie Alert-FUN-014731-Chimes).

- This includes the Chime Client sending the turn signal Tic and Toc sounds in the IPC_Chime and IPC_Chime2 message. The SPSS already supports this so this is just a chime arbitrator spec update. The Chime Client (VMCU) must take care to make sure the turn signal visual indicator and audio chime are synchronized as much as possible.

Note:

- ICI stands for Integrated Cluster & IAHU (AHU and System Master)
- If the sequences diagrams reference CAN that should be ignored as the sequence diagrams should be considered at the logical level (ie not network dependent)
- In this spec if legacy requirements reference AHU then this references the SYNC 4.2 module or chime generator (ie CCPU).
- In this spec if legacy requirements reference Cluster then this references the SYNC 4.2 module or chime client / chime master (ie VMCU).

3.1.2 Deployment chimes variant 2 (for reference only)

Logical Class	SYNC 4.2	DSP AMP variant 2	SYNC 4.2 VMCU (M4 core)	SYNC 4.2 CCPU (core A QNX)
Chime Client (Chime Master)	x		x	
Chime Generator	x			x
Chime Audio Source	x			x
Chime Audio Attenuation	x	x (optional – see DSP variant 2 chime audio attenuation note below)		x
Back-up chime generator (CCPU down) / Early chime generator	x		x	
Back-up chime generator when CCPU powered up	x			x
Chime Arbitrator	x		x	
Chime Diagnostics	x	x		

x – represents functionality supported by that module
blank – represents functionality not supported by that module

Note:

- The implementation guide class descriptions are the final say between what is the Client and Server. This is for reference only and for any conflict with implementation guide then the implementation guide will takes precedence. Bring any conflicts to Ford's attention
- DSP variant 2 chime audio attenuation: only applies if the DSP AMP variant 2 ever generates audio itself. Doesn't apply if just passing audio from the SYNC 4.2 module (SYNC 4.2 would be responsible for muting its outputs)
- The Chime Arbitrator is part of the Chime Client/Chime Masters responsibility and is captured in a separate Chime Arbitrator specification

3.1.3 Chime Activation Requirements

3.1.3.1 ALERTv2-SR-REQ-376339/A-Infotainment System Chime Source - Variant 2

For SYNC 4.2 the Cluster VMCU shall use the IPC_Infotainment : Chime_Source signal to indicate if the Infotainment System (ie all the vehicle speakers) or Cluster (ie back-up speakers) is currently the chime source responsible for playing chimes. The full Infotainment System (including the DSP AMPv2) shall NOT be the chime source unless the IPC_Infotainment.St() message has signal 'Chime_Source = Infotainment_System'.

For SYNC 4.2 when Chime_Source = Cluster:

- If AHU_Chime_Supported = Supported then the CCPU shall control the back-up speaker(s)
- If AHU_Chime_Supported = Inactive or Not_Supported then the VMCU shall control the back-up speaker(s).

Note:

- for SYNC 4.2 hardware (at the time this requirement was written) the VMCU and CCPU share the same AMP. If the VMCU has control that means there is no infotainment audio (ex AM/FM, USB, Phone Call…) so the CCPU back-up should be used whenever possible.

The CCPU shall tell the VMCU if it is in full speaker mode or back-up speaker mode so the VMCU can set the Chime_Source signal correctly.

- Note: modules like the DSP AMP variant 2 still use the Chime_Source signal under certain conditions.

3.1.3.2 ALERT-SR-REQ-014733/C-Infotainment Components Power Mode signals (TcSE ROIN-40352-1)

The Infotainment Chime Components shall be capable of producing chimes from Chime Requests (IPC_Chime / IPC_Chime2) when:

'Power_Up_Chime_Modules == Active' AND the Infotainment System is the Chime Source 'IPC_Infotainment.St() : Chime_Source == Infotainment_System'.

The infotainment components responsible for chimes shall be capable of producing chimes regardless of the HMIAudioMode status when Chimes is enabled through the Infotainment System.

3.1.3.3 ALERTv2-SR-REQ-361635/A-Chime Power Mode signal usage - variant 2

The Cluster shall wake and keep up the Infotainment bus (if not already awake) and set 'Power_Up_Chime_Modules = Active' for predictive triggers (ex. open door) as defined in the Cluster engineering specification or predictive triggers called out in the P06 power management spec.

- The SYNC 4.2 P06 power management spec calls out predictive triggers when the SYNC CCPU is powered up such as door unlock, door open and approach detection.

When a predictive trigger does occur then the minimum time the signal 'Power_Up_Chime_Modules' is set equal to 'Active' is 180 seconds.

- An exception would be for error states such as the AHU setting AHU_Chime_Supported = Not_Supported where the Cluster would set Power_Up_Chime_Modules = Inactive when it receives Not_Supported.

When the predictive triggers are no longer valid then the signal 'Power_Up_Chime_Modules = Inactive'.

3.1.3.4 ALERT-SR-REQ-014735/E-Power-up time for infotainment components (TcSE ROIN-40354-4)

The infotainment chime components shall be capable of producing chime audio within 2.0 seconds of 'Power_Up_Chime_Modules = Active'.

For supporting time to first infotainment chime audio reference the Infotainment Diagnostic Specification (IDS) for chime diagnostics start-up using the Power_Up_Chime_Module signal.

3.1.3.5 ALERT-SR-REQ-014737/C-Timing for single Chime Source module to produce Chime audio (TcSE ROIN-40356-3)

When the Chime Generator and Chime Audio Source are the same Chime Source module then the Chime Source module shall start producing chime audio within 70 msec of receiving the new chime request message.

3.1.3.6 ALERT-SR-REQ-014741/J-Cluster Chime Requests (TcSE ROIN-40360-1)

Once the Cluster receives a chime request (from the vehicle network) it shall process and transmit to the Infotainment System within 60 msec the applicable IPC_Chime message to the infotainment system with the chime signal parameters set.

If there is a currently playing chime the Cluster shall determine the priority of the currently playing chime to see if it should be overwritten by the new chime request. If the new chime request has a higher priority than a currently playing fixed repetitive chime then the Cluster may want to monitor the AHU_Chime_Active signal to see if the new chime can be delayed until the current chime is complete or if it should immediately be overwritten.

The Cluster shall set all the signals in the IPC_Chime (1 or 2) messages and keep all the signals populated and unchanged as long as the chime is active (exception OFF_Time_Btwn_Chime signal). Only when the Chime is done should the IPC_Chime signals be set to No_Chime and Inactive.

The Cluster shall send a No_Chime for the currently playing chime before sending any new chime requests with the directionality for the same speakers. See sequence diagrams for examples of when chime requests are sent by the Cluster.

Note: If anything in the Cluster chime arbitrator spec contradicts what is in the chime section of the SPSS it should be brought to Ford's attention.

Multiple Chime Requests for speakers with the same directionality:
If the AHU is playing a chime (ex AHU_Chime_Active = Chime_Y) and if the Cluster needs to end the chime and play a new chime using the same speakers/directionality, then once the Cluster sends the No_Chime chime request to end Chime_Y the Cluster shall wait for the AHU to respond back with "AHU_Chime_Active = No_Chime" before making the new chime request. The AHU shall respond back to the Cluster No_Chime request within 75 msec with a No_Chime.

- Note: because of legacy Clusters the AHU cannot assume the Cluster will always wait for the AHU No_Chime response as stated above if two chime requests occur on different alert channels (ie IPC_Chime/IPC_Chime2). For the AHU's if they receive a cluster chime request (Chime_Y) on one Alert channel and then receive the No_Chime request on the same Alert channel the Cluster could send a chime request for the same speakers/directionality on a different alert channel before the AHU responds back with a No_Chime and the AHU still has to support. See sequence diagrams with examples.
    - This note does not apply to APIM 4.2 since this would no longer be considered a legacy Cluster.

If the AHU is not playing a chime (AHU_Chime_Active = No_Chime) and if the Cluster sends a chime request such as IPC_Chime(Chime = Chime_X) and then the Cluster quickly sends a No_Chime before the AHU responds to the original Chime_X request with AHU_Chime_Active = Chime_X then the Cluster shall not send a new chime request until:

- The AHU responded back with AHU_Chime_Active = Chime_X and then AHU_Chime_Active = No_Chime, OR
- If the AHU does not respond back to the original Chime_X response (stays at AHU_Chime_Active = No_Chime) the Cluster shall wait 75 msec before making any new chime requests

3.1.3.7 ALERT-SR-REQ-014742/C-Chime_Vol_Level signal (TcSE ROIN-40361-3)

The Chime Audio Source stores the volume levels for each of the individual chimes. The Chime Audio Source will use the Chime_Vol_Level signal from the Cluster to determine how much lower the chime should be played from the stored volume level.

If the stored chime volume level in the chime audio source is to be played at 100% of the stored volume level then the Cluster shall set the signal Chime_Vol_Level = Inactive.

Once a chime is selected by the Cluster and broadcast on the infotainment bus the Cluster shall not change the selected chime volume level until the chime has ended or until there is a new chime activated.

3.1.3.8 ALERT-SR-REQ-014743/C-Chime Mixing (TcSE ROIN-40362-1)

The Chime Audio Source shall be capable of mixing the Chimes together with the main audio source as defined in the applicable Chime Audio Source component engineering specifications.

3.1.3.9 ALERT-SR-REQ-368145/B- Chime_Not_Recognized signal - variant 2

Variant 2 of the Chime_Not_Recognized requirement applies when the Cluster chime arbitrator (VMCU) and Chime Generator functionality (CCPU) are internal to the same module sharing the same audio files. If for some reason the Cluster software requests a chime that the Chime Generator (CCPU) does not recognize then the Chime Generator shall set Chime_Not_Recognized as stated below. The VMCU could perform error handling (ex retry new request) or not play the chime at all.

AHU Chime Generator:
If the Chime Generator receives a command from the Cluster to play a particular chime that the Chime Generator does not recognize (Chime_X were X is unknown) then the Chime Generator shall send the 'XXX_Chime_Not_Recognized = Chime_Not_Recognized' signal within 75msec of receiving the chime command.

Whenever the Chime Generator sets the signal Chime_Not_Recognized = Chime_Not_Recognized then at the same time in the same message the AHU_Chime_Active signal will be used to say what Chime the Chime Generator cannot play.

Note: for SYNC 4.2 if the Cluster VMCU played the chime when it received the Chime_Not_Recognized signal then there would be no infotainment audio (ex AM/FM, USB, Phone call audio) based on the hardware of the SYNC 4.2 module at the time this spec was written (CCPU couldn't play audio). This would need to be taken into consideration for chime error handling if the chime should be played on the VMCU or not.

3.1.3.10 ALERT-SR-REQ-014746/D-OFF_Time_Between_Chime signal (TcSE ROIN-40366-1)

The Cluster shall tell the Chime Generator how long there is no chime audio between playing a particular chime sound using the OFF_Time_Btwn_Chime signal (ex. pauses between repeating Chime_17 Reverse Park Aid sound). For the same chime this value may change.

Within a particular Chime Request from the Cluster the OFF_Time_Btwn_Chime signal could change for the same chime but this should not change the state of the Chime Audio Source Alert input channel settings (ex volume, directionality). While the OFF_Time_Btwn_Chime signal is changing for the same chime the Alert_Channel would remain Initialized_For_Chimes.

For example for a reverse park aid chime as the vehicle backs up the OFF_Time_Btwn_Chime signal could change to increase the beep rate by making the OFF_Time_Btwn_Chime time a smaller value.

Ex:
1. RPA Chime event occurs
2. The Cluster sends IPC_Chime/IPC_Chime2 (Chime = RPA_Chime; Chime_Occurance = Continuous; OFF_Time_Btwn_Chime = Zone1 Rate…)
3. The AHU starts playing the RPA chimes and sends AHU_Alert (AHU_Chime_Active = RPA_Chime).

4. The vehicle goes into another zone and sends IPC_Chime/IPC_Chime2 (Chime = RPA_Chime; Chime_Occurance = Continuous; OFF_Time_Btwn_Chime = Zone2 Rate…) without sending a No_Chime and only changes the OFF_Time_Btwn_Chime signal.
5. The AHU plays the RPA chimes at the rate for Zone 2 and continues to send (AHU_Chime_Active = RPA_Chime).

Note: in the above example both the Cluster and AHU did not send a No_Chime when only the OFF_Time_Btwn_Chime signal changed.

3.1.3.11 ALERT-SR-REQ-014747/J-Chime_Time_Criticality = Criticality_High (TcSE ROIN-40367-1)

The Cluster shall set 'Chime_Time_Criticality = Criticality_High' if a new Chime is a chime that needs the currently playing chime sound (ex Chime_8 Ford DNA B) to be immediately ended so the new Chime can be played as quickly as possible (ex. FCW).

If the Chime Generator received a Criticality_High Cluster No_Chime Chime Request to end the currently playing chime then if the chime sound is playing (ex DNA B) the Chime Generator shall immediately end playing that chime sound. See AHU hardware spec for what is the longest it can take to end a chime sound with no distortions or pops (ex 10 msec or whatever defined in AHU hardware spec).

If the Chime Generator received a criticality low No_Chime request from the Cluster to end the current chime and while the Chime Generator is still playing the current chime sound (ex AHU_Chime_Active = Chime_X) if the Chime Generator then receives a criticality_high chime request for the same speakers from the Cluster then the Chime Generator shall immediately end the current chime (ie immediately end Chime_X sound). See sequence diagrams for detailed examples.

Ex.
Pre-Condition:
Infotainment Chime Generator and Chime Audio Source are playing a continuous chime.

Event:
Chime Clients ends the chime with Criticality_High No Chime event. Chime Client sends IPC_Chime (Chime = No Chime, Chime_Time_Criticality = Criticality_High, Chime_Vol_level = Inactive; Chime_Directionality = Inactive; OFF_Time_Btwn_Chime = Inactive; Chime_Occurence = Inactive)

Post-Condition:
The Chime Generator ends it currently playing Chime_X sound (ex DNB) before it is finished and immediately sets AHU_Chime_Active = No Chime

3.1.3.12 ALERT-SR-REQ-014748/I-Chime_Time_Criticality = Criticality_Low (TcSE ROIN-40368-1)

The Cluster shall set 'Chime_Time_Criticality = Criticality Low' if a new Chime is a chime that can wait for the currently playing chime to finish playing the chime sound (ex. finish Ford DNA B sound) before playing the new chime.

If the Chime Generator receives a Criticality_Low Cluster No_Chime request to end the current chime then if the Chime Generator is playing a chime sound (ex Ford DNA B sound) it shall finish playing the chime sound and then end the chime (ie set AHU_Chime_Active = No_Chime).

If the Chime Generator received a criticality_low Cluster chime request "Chime = No_Chime" to end the current chime (ex Chime_X) and while the Chime Generator is still playing the previous chime sound (ex AHU_Chime_Active = Chime_X) the Chime Generator received a new low criticality Chime request for the same speakers then the AHU / DSP AMP shall start to play the new chime after playing the currently playing chime sound finishes (the AHU/DSP AMP shall play the new chime as long as the Cluster has not sent the No chime for the new chime request (ex. IPC_Chime/IPC_Chime2 (Chime = Chime_New → Chime = No_Chime)) by the time the AHU/DSP AMP finishes playing the old chime sound (Chime_X)). See sequence diagrams for examples.

Example 1: If the Chime_Occurance was set to 8 repetitions of the Chime_X sound (ex Ford DNA B) and the 3rd repetition Chime_X sound is currently being played when the new Chime_Y is requested with Chime_Time_Criticality = Criticality Low

then the Chime Generator would finish the 3rd repetition of the Chime_X sound and keep AHU_Chime_Active = Chime_X until finished but not play the remaining 5 repetitions but instead switch to the new chime. See sequence diagrams for detailed examples.

Note: The Cluster can always just wait for the chime to complete it's repetitions by monitoring the AHU_Chime_Active = Chime_X signal to avoid having to interrupt a chime.

Example 2.
Pre-Condition:
Infotainment Chime Generator and Chime Audio Source are playing a continuous chime (AHU_Chime_Active = Chime_X).

Event:
Chime Clients ends the chime with Criticality_Low No Chime event. Chime Client sends IPC_Chime (Chime = No Chime, Chime_Time_Criticality = Criticality_Low, Chime_Vol_level = Inactive; Chime_Directionality = Inactive; OFF_Time_Btwn_Chime = Inactive; Chime_Occurence = Inactive)

Post-Condition:
The Chime Generator ends it currently playing Chime_X sound (ex DNA B) after the Chime Generator is finished producing the Chime_X sound and sets ChimeGenerator_Chime_Active = No_Chime after it is done producing the sound

3.1.3.13 ALERTv2-SR-REQ-348582/B-Chime_Directionality signal - variant 2

This variant 2 of the Chime Directionality requirement applies if the Chime Generator supports Directionality All chime along with another chime concurrently:

- The Chime Generator, if it uses one polyphonic and one simple chime generator, shall follow variant 1 of this requirement (Alert-REQ-014749-Chime_Directionality signal) which does not allow directionality All chime along with another chime.
- The Chime Generator, if it uses stereo digital media chimes for all chimes, then this requirement shall be followed (allow a chime with directionality All and another Front or Rear chime to be played at the same time).
- APIM 4.1 (integrated AHU) and APIM 4.2 (integrated AHU and Cluster) shall follow this requirement

Chime Client making Directionality All chime request along with another chime request:
The Chime Client (ex Cluster) shall not request a directionality All chime request with another chime request (ie Front or Rear) at the same time until paired with a Chime Generator and Chime Audio Source that supports this requirement (directionality All plus another Front or Rear chime). If the Cluster does request this on a chime generator that does not support this requirement, then the AHU would set Chime_Not_Recognized for the second chime and it would be played by the Cluster.

The Cluster shall tell the Chime Generator and Chime Audio Source what speakers to play the chime through using the Chime_Directionality signal.

- Note throughout this requirement what applies to Chime_Directionality signal also applies to the Chime_Directionality2 signal

Chime_Directionality signal:

- If Chime_Directionality = Front then the requested chime plays out of the front speakers
- If Chime_Directionality = Rear then the requested chime plays out of the rear speakers
- If Chime_Directionality = All then the requested chime plays out of all speakers

Allowable Combinations that the Cluster can send:
1. 1 chime Front speakers only (ex Chime_Directionality = Front)
2. 1 chime Rear speakers only (ex Chime_Directionality = Rear)
3. 1 chime out for Front speakers and one chime out of Rear speakers at the same time
4. 1 chime out of All speakers (ex Chime_Directionality = All) and no other chimes
5. 1 chime out of All speakers and one chime out of the Front speakers at the same time
6. 1 chime out of All speakers and one chime out of the Rear speakers at the same time (ex Chime_Directionality = All and Chime_Directionality2 = Rear)

Vehicle with Front Speakers Only (no rear speakers):

For this vehicle set-up Chime_Directionality = Rear and Chime_Directionality = All would be played out of the front speakers by the Chime Generator / Chime Audio Source.

Error Handing if chime requests that are not an allowable combination:
If the chime generator receives 2 chime requests with the same directionality (ie Chime_Directionality & Chime_Directionality2 both equal Front, both equal to Rear, or both equal to All) then the Chime Generator / Chime Audio Source shall follow the request and play both chimes out of the same speakers with the same directionality.

3.1.3.14 ALERT-SR-REQ-014750/C-Chime Audio Attenuation (TcSE ROIN-40370-1)

While chime(s) are active the Cluster shall tell the Chime Audio Source if the infotainment audio will be attenuated, muted, or if there will be no effect on non-chime infotainment audio via the 'IPC_Infotainment.St() : Attn_Info_Audio' signal.

3.1.3.15 ALERT-SR-REQ-014752/C- Chime Active signal from Chime Generator (TcSE ROIN-40394-3)

The Chime Generator shall indicate what chime it is playing to the chime components (ex. Cluster, Chime Audio Source) via the '_Chime_Active = Chime_X' signal. This event-periodic signal will be updated on event when the Chime Generator starts or stops playing a chime.

When there is no chime being played then the '_Chime_Active' signal shall equal 'No_Chime'. Note: this does not mean that in between chime sounds (Off_Time_Btwn_Chime) that 'No_Chime' is set.

When the Chime_Occurance signal is set to a fixed repetition rate the Chime Generator shall not set the _Chime_Active signal equal to "No_Chime" until the last repetition and chime sound is finished playing. Until the last chime sound is played the _Chime_Active shall be set to the current chime.

3.1.3.16 ALERT-SR-REQ-014753/D-Chime Occurance signal (TcSE ROIN-40395-1)

The Cluster shall tell the Chime Generator the Occurrence of the chime to be played using the Chime_Occurance signal.

-- If Chime_Occurence is set to 1, 2, 3… repetitions then the chime shall be played 1, 2, 3… times respectively and then the chime shall end (unless repetition ended early by the Cluster with IPC_Chime : Chime = No_Chime). The Cluster shall know that the chime has ended by looking at what the Chime Generator signal 'XXX_Chime_Active' is set equal to. If it's set to 'No_Chime' then the chime has ended.

-- If Chime_Occurence is set to 'Continuous' then the Chime shall play as long as the Chime_X signal in 'IPC_Chime : Chime = Chime_X' doesn't change.

- When the chime is turned off (IPC_Chime : Chime = No Chime') then the currently playing chime will end. It shall either finish playing the chime or immediately end depending on the Chime_Time_Criticality signal state.

3.1.3.17 ALERT-SR-REQ-014754/C-Chime / Prompt / Beep Prioritization (TcSE ROIN-40432-5)

The Chime Audio Source shall be capable of mixing a chime and prompt together at the same time. If a beep event occurs while both a chime and prompt are active then the chime audio source shall mix in the beep if it is capable otherwise the beep shall not be heard. A chime has higher priority than a beep.

The Prompt Generator can monitor the IPC_Infotianment : Attn_Info_Audio / New_Attn_Event signal to see if the prompt is muted or attenuated. This could be used by the prompt generator to avoid prompts seeming to be starting mid-sentence to the user because of a muting event caused by a chime.

Alert Channel 1 between the Chime Generator and Chime Audio Source is used for Chime1 as indicated in the IPC_Chime.St message

Alert Channel 2 between the Chime Generator and Chime Audio Source is used for Chime2 as indicated in the IPC_Chime2.St message.

3.1.3.18 ALERT-SR-REQ-014755/H-Chime ID Assignments (TcSE ROIN-167427-3)

CAN	Chime_X	Chime Sound ID
0x1	Chime_0	No Chime
0x2	Chime_1	Turn Signal (Tic)
0x3	Chime_2	Turn Signal (Toc)
0x4	Chime_3	1.0 Second Chime
0x5	Chime_4	0.5 Second Chime
0x6	Chime_5	0.25 Sec Chime
0x7	Chime_6	1 Sec Tone (1KHz Alert)
0x8	Chime_7	0.1 Sec Chime
0x9	Chime_8	Ford DNA Chime B (Soft Warning)
0xA	Chime_9	Ford DNA Chime C (Hard Warning)
0xB	Chime_10	Ford DNA Chime D (Non-Critical Alert) - Info
0xC	Chime_11	Ford DNA "B" shortened to 0.5 sec
0xD	Chime_12	Perimeter Warn. Chime A
0xE	Chime_13	Perimeter Warn. Chime B
0xF	Chime_14	Perimeter Warn. Chime C
0x10	Chime_15	Cross-Traffic Alert (CTA)
0x11	Chime_16	Forward Park Aid
0x12	Chime_17	Reverse Park Aid
0x13	Chime_18	Lincoln DNA Chime B (Soft Warning)
0x14	Chime_19	Lincoln DNA Chime C (Hard Warning)
0x15	Chime_20	Lincoln DNA Inf Chime D (Non-Critical Alert)- Info
0x16	Chime_21	Lincoln DNA "B" shortened to 0.5 sec
0x17	Chime_22	ACC-High, and FCW
0x18	Chime_23	Lane Departure Warning (LDW)
0x19	Chime_24	Push Button
0x1A	Chime_25	Beltminder A
0x1B	Chime_26	Beltminder B
0x1C	Chime_27	RPA Continuous
0x1D	Chime_28	FPA Continuous
0x1E	Chime_29	Power Liftgate (POT) / Power Sliding Door
0x1F	Chime_30	Chime_30
0x20	Chime_31	Chime_31
0x21	Chime_32	Chime_32
0x22	Chime_33	Chime_33
0x23	Chime_34	Chime_34
0x24	Chime_35	Digital Audio Chime file (stereo chime)
0x25	Chime_36	Digital Audio Chime file (stereo chime)
0x26	Chime_37	Digital Audio Chime file (stereo chime)
0x27	Chime_38	Digital Audio Chime file (stereo chime)
0x28	Chime_39	Digital Audio Chime file (stereo chime)
0x29	Chime_40	Digital Audio Chime file (stereo chime)
0x2A	Chime_41	Digital Audio Chime file (stereo chime)
0x2B	Chime_42	Digital Audio Chime file (stereo chime)
0x2C	Chime_43	Digital Audio Chime file (stereo chime)
0x2D	Chime_44	Digital Audio Chime file (stereo chime)
0x2E	Chime_45	Chime_45 (Heavy Truck Lane Departure Warning)
0x2F	Chime_46	Reserved
0x30	Chime_47	Chime_47 (ex Hands on Wheel 1 - Highway Assist)
0x31	Chime_48	Chime_48 (ex Hands on Wheel 2 – Highway Assist)
0x32	Chime_49	Chime_49 (ex Hands on Wheel 3 – Highway Assist)
0x33	Chime_50	Digital Audio Chime file (stereo chime) – (ex BEV ON)

0x34	Chime_51	Digital Audio Chime file (stereo chime) – (ex BEV OFF)

Note: Chime_X are referred to as Chime Sound in a number of requirements

3.1.3.19 ALERT-SR-REQ-237862/D-Chime_ID signal

Note throughout this Chime_ID signal requirement when the SPSS specifies what to do for the ChimeID_No_Rq / ChimeID_No_Stat signals the same requirements apply to Chime2ID_No_Rq / Chime2ID_No_Stat signals.

- IPC_Chime(ChimeID_No_Rq) corresponds to AHU_Alert(ChimeID_No_Stat)
- IPC_Chime2(Chime2ID_No_Rq) corresponds to AHU_Alert(Chime2ID_No_Stat)

Note: The Chime_ID signals were added per the Cluster Ford team since they are needed to identify chime active responses back from the AHU to make sure the Cluster Chime Request and AHU chime active responses (ie AHU_Chime_Active) were always talking about the same chime (chime = all the signals in IPC_Chime/IPC_Chime2). This was identified as a potential Cluster issue for multiple chime requests where Chime = Chime_X for a sound which could be used for multiple chimes.

- This Chime ID requirement needs to be supported when Adjustable Chimes is supported. Once this requirement is supported in the AHU software then the Chime ID signal shall continue to be supported by the AHU regardless whether adjustable chimes is supported or not.

Cluster Chime Client:
The Cluster / Chime Client shall set the signal ChimeID_No_Rq equal to an ID value to uniquely identify all chime requests to the infotainment system (see chime arbitrator specs for details on implementing this in the Cluster).

Once the Cluster sets ChimeID_No_Rq equal to an ID value for a chime request it shall hold this value for the entire time this chime is active in the IPC_Chime chime request message.

- Note: signal OFF_Time_Btwn_Chime is allowed to change and still be considered the same chime

The Cluster shall be backwards compatible with AHU modules that don't support the Chime_ID signals (sourced to earlier versions of the Alerts SPSS). If not supported the ChimeID_No_Stat from the AHU should always be set to 0x0 Inactive.

Some chime sequence diagrams where updated to show the ChimeID_No_Rq and ChimeID_No_Stat signals. Reference those sequence diagrams for examples of those signals being used.

AHU Chime Generator:
The AHU Chime Generator shall set the signal ChimeID_No_Stat equal to the value in the Clusters ChimeID_No_Rq signal when playing that particular chime (IPC_Chime(Chime = Chime_X; ChimeID_No_Rq = Y…)). The AHU shall continue to send the same ChimeID_No_Stat value from the Cluster ChimeID_No_Rq chime request signal until the AHU is done playing the chime (either AHU finishes playing or the Cluster ends the chime). The AHU shall never change the ChimeID_No_Stat signal while playing a chime regardless what the Cluster is sending in its ChimeID_No_Rq signal.

When the AHU sets AHU_Chime_Active = No_Chime the ChimeID_No_Stat signal shall be set to 0x1 (ID used for No_Chime).

Some chime sequence diagrams where updated to show the ChimeID_No_Rq and ChimeID_No_Stat signals. Reference those sequence diagrams for examples of those signals being used.

In the case where Cluster doesn't support Chime ID signal but the AHU does then the AHU shall just repeat whatever is in the ChimeID_No_Rq signal. The ChimeID_No_Rq signal does not have any effect on the AHU chime audio itself so the AHU is backwards compatible with Clusters that don't support Chime ID signals..

- If the Cluster keeps ChimeID_No_Rq = 0x0 Inactive while chimes is active (because it doesn't support chimes) then the AHU shall set ChimeID_No_Stat = 0x0 Inactive.
  - Exception: If simpler on the AHU software when there is no chime the AHU could still set ChimeID_No_Stat = 0x01 ID for No Chime.

Example1 (AHU ends chime when finished playing it):

1. Cluster sends IPC_Chime (Chime = Chime_X; ChimeID_No_Rq = Z; Chime_Occurence = 5 repetitions;…)
2. AHU plays the chime and sends AHU_Alert(AHU_Chime_Active = Chime_X; ChimeID_No_Stat = Z) while playing the chime

3. AHU finishes the 5th repetition and sends AHU_Alert(AHU_Chime_Active = No_Chime; ChimeID_No_Stat = 0x01 ID for AHU No Chime).

Example 2 (Cluster ends a chime before the AHU is finished playing it):
1. Cluster sends IPC_Chime (Chime = Chime_X; ChimeID_No_Rq = Z; Chime_Occurence = 5 repetitions;…)
2. AHU starts playing the chime and sends AHU_Alert(AHU_Chime_Active = Chime_X; ChimeID_No_Stat = Z)
3. Cluster ends the chime before the AHU finishes playing the 2nd repetition chime sound and sends IPC_Chime(Chime = No_Chime; ChimeID_No_Rq = don't care for AHU; Chime_Time_Criticality = Criticality_Low)
4. AHU sends AHU_Alert(AHU_Chime_Active = Chime_X; ChimeID_No_Stat = Z) until the second repetition Chime_X sound is complete
5. When AHU finishes the 2nd repetition it sends AHU_Alert(AHU_Chime_Active = No_Chime; ChimeID_No_Stat = 0x01 ID for AHU No Chime)

DSP AMP chime audio source:
No impact to the DSP AMP

3.1.3.20 ALERT-SR-REQ-052682/J-Chime Requests while the infotainment system is muted during a cold crank

During a cold crank event infotainment chimes are muted by the infotainment system. If the Cluster needs to have a chime played during a cold crank while the infotainment system is muted, then the Cluster would have to play the chime through the Cluster.

After a crank event ends the Chime Generator shall unmute and be capable of playing the chimes no later than 1000 msec after the crank event ends.
- The Crank event ending for both the AHU, DSP AMP and DSP AMPv2 is defined in requirements "STMGNTv2-GFUN-202153-1-Crank, Front System ON" for CGEA 1.3 and "STMGNT-FUN-014666-Crank, Front System ON" for CGEA 1.2 / C1MCA.
- Reference the EMC specification(s) for the worst case cold crank voltage profile that needs to be supported (ex EMC requirement CI-230).

After a crank event ends but before the Chime Generator has unmuted if the chime generator receives a chime request then:
- the Chime Generator shall still respond to IPC_Chime/IPC_Chime2 chime request messages and behave as if it is playing the chime (ex chime generator responds with AHU_Alert : AHU_Chime_Active = Chime_X)
- as soon as the Chime Generator is unmuted the chime will be playing as if the chime began playing with the initial chime request (exception seat belt chime). For example, if the chime had 3 repetitions then by the time the Chime Generator is unmuted it may be on the 2nd or 3rd repetition.

Exception to Chime Generator requirement above - Seat Belt Chime:
After a crank event ends and the Chime Generator has not yet unmuted if the chime generator had received a Seat Belt Chime request from the Cluster while the network signal said crank (if message was not lost with crank voltage dip) or after crank ends but before the AHU / DSP AMP unmuted then (ie IPC_Chime /IPC_Chime2 = "Seat Belt Chime") then:
1. the chime generator shall update the AHU_Chime_Active/AHU_Chime_Active2 signal to Seatbelt Chime when it received the IPC_Chime request (follow sequence diagram for when DSP AMP present), and
   - AHU Only (No DSP AMP):
     - then the chime generator delays playing the seat belt chime until after the chime generator is unmuted as long as:
       i. when the unmute first occurs the Cluster is still sending the seat belt chime request message (IPC_Chime = "Seat Belt Chime")
       ii. and the Cluster has not already ended the chime (ie IPC_Chime : Chime = No_Chime).

       The Chime Generator shall start playing the seat belt chime no later than 20 msec after the unmute occurs.
   - AHU with DSP AMP or DSP AMPv2 combination:
     - then the chime generator delays playing the seat belt chime until 950 msec +/- 10 msec after the crank event ends as defined in Station Management SPSS crank requirements (STMGNT-FUN-014666, STMGNT-FUN-202153) as long as:
       - when the unmute first occurs the Cluster is still sending the seat belt chime request message (IPC_Chime = "Seat Belt Chime")

The information contained in this document is Proprietary to Ford Motor Company.

- and the Cluster has not already ended the chime (ie IPC_Chime : Chime = No_Chime).

The IPC_Chime / IPC_Chime2 seat belt chime is active when the following signals are set:

Ford Seat Belt Chime:
IPC_Chime/IPC_Chime2 (Chime = 0x0B Chime_10, Chime_Vol_Levl = 0x0 100% stored volume; Chime_Directionality = 0x2 Front; Chime_Time_Criticality = 0x1 Criticality_High; OFF_Time_Btwn_Chime = 0x1 Continuously; Chime_Occurance = 0x4 4 repetitions)

Lincoln Seat Belt Chime:
IPC_Chime/IPC_Chime2 (Chime = 0x15 Chime_20, Chime_Vol_Levl = 0x0 100% stored volume; Chime_Directionality = 0x2 Front; Chime_Time_Criticality = 0x1 Criticality_High; OFF_Time_Btwn_Chime = 0x1 Continuously; Chime_Occurance = 0x4 4 repetitions)

Example – Seat belt chime request after a crank event ended but before the AHU is unmuted (No DSP AMP present):
1. Crank event ends
2. The Cluster sends IPC_Chime (Chime = 0x0B Chime_10, Chime_Vol_Levl = 0x0 100% stored volume; Chime_Directionality = 0x2 Front; Chime_Time_Criticality = 0x1 Criticality_High; OFF_Time_Btwn_Chime = 0x1 Continuously; Chime_Occurance = 0x4 4 repetitions)
3. The AHU responds within 75 msec with AHU_Alert.st (AHU_Chime_Active = 0xA Chime_10)
4. 600 msec (could be any value under 1000 msec) after the crank event ended the AHU unmutes and starts playing the first seat belt chime sound and plays all 4 repetitions (unless Cluster ends the chime before finished).

3.1.3.21 ALERT-SR-REQ-348595/A-DSP AMP variant 2 chime functionality

The DSP AMPv2 (ie DSP AMP variant 2) does not support IPC_Chime / IPC_Chime2 chime requests since the DSP AMPv2 is not the Chime Generator or Chime Audio Source.

The Chime Generator / Chime Audio Source for the A2B chime outputs to the DSP AMPv2 is responsible for muting and unmuting between chime events.

The DSP AMPv2 does support chime diagnostics as defined in the IDS and applicable requirements in this chime spec for the Chime_Supported signal.

The DSP AMPv2 module shall mute its A2B chime inputs when:

- the Chime_Supported signal for DSP_AMPv2 is not set to Supported, OR
- the Power_Up_Chime_Modlue signal is set to Inactive

When the DSP AMPv2 Chime_Supported signal is set to Supported with Power_Up_Chime_Modules = Active then the DSP AMPv2 shall unmute its A2B chime inputs. The DSP AMPv2 shall unmute its A2B chime inputs before setting Chime_Supported to Supported.

The DSP AMPv2 is mutually exclusive with the DSP AMP. In this chime specification when it says DSP AMP it does not apply to the DSP AMPv2.

The Chime Generator shall send A2B Chime Audio as follows:

- Chime_Directionality = Front then front chime audio will use A2B Stereo Chime Front
- Chime_Directionality = Rear then rear chime audio will use A2B Stereo Chime Rear
- Chime_Directionality = All then chimes on front and rear speakers would use A2B Stereo Chime Front and A2B Stereo Chime Rear

Note: above used Chime_Directinality signal but this also applies to Chime_Directionality2 signal. The DSP AMP variant 2 does not look at the Chime_Directionality signal.

3.1.3.22 ALERT-SR-REQ-361687/A-Speed Compensated Volume Chimes

SYNC 4.2 CCPU shall support all infotainment chimes adjusting their volume based on the speed of the vehicle.

Speed compensated volume chimes shall be configurable end of line such that speed compensated volume chimes could be disabled EOL.

See core audio requirement TBD for details (tied to current SCV requirement TBD)

3.1.3.23 ALERT-SR-REQ-361909/A-VMCU Chime Generator

The VMCU back-up / early chime generator shall be capable of playing all chimes out of the back-up speaker(s).

- Note: this does not require the audio quality to be as good as when through the CCPU

The VMCU back-up / early chime generator shall be responsible for playing the chimes when the CCPU cannot play the chimes.

TBD – Does the VMCU play through front and REAR back-up speaker if the CCPU takes a while to boot up (ex 5 - 15 seconds)? If so, how would that work when there was an external amp?

Note: See "Alert-REQ-361651-Chime Error States / Fault handling – variant 2" for details of fault handling and using the AHU_Chime_Supported signal regarding the VMCU being the back-up chime generator

3.1.4 Chimes Error Management Requirements

3.1.4.1 ALERTv2-SR-REQ-361649/B-Loss of communication with Chime module - variant 2

The Cluster VMCU shall monitor the AHU (CCPU) internal to SYNC 4.2 to make sure it can play chimes. The VMCU shall monitor the CCPU and if the CCPU is no longer detected (ex rebooted/reset) while chimes are through the infotainment system then the VMCU shall play the chimes through the back-up speaker(s) and set the signal 'IPC_Infotainment.St() : Chime_Source = Cluster' and 'Power_Up_Chime_Modules = Inactive.

If the Chime Generator loses communication with the DSP AMPv2 module for more than 5 seconds, then:

- the Chime Generator (ie CCPU) shall play chimes through the back-up speaker(s),
- or if for some reason it cannot play chimes through the back-up speaker(s) then set AHU_Chime_Supported = Not Supported so the Cluster (VMCU) plays the chimes through the back-up speaker(s).

If the CCPU is detected and working again with AHU_Chime_Supported = Supported then the chimes shall be through the CCPU (either full infotainment system or back-up).

3.1.4.2 ALERTv2-SR-REQ-361651/B-Chime Error States / Fault handling - variant 2

If the Chime Generator (ie CCPU) sets its signal 'AHU_Chime_Supported' equal to 'Not_Supported' for a particular configuration then the Cluster (ie VMCU) shall become the Chime Generator and Chime Audio Source and sets the 'IPC_Infotainment.St() : Chime_Source = Cluster' and Power_Up_Chime_Modules = Inactive. When the Cluster is the Chime Generator and Chime Audio Source it shall play the chimes out of the back-up speaker(s).

The Chime Generator (ie CCPU) shall monitor the DSP AMPv2 DSP_Chimes_Supported network signal. If the DSP_Chimes_Supported signal equals Not_Supported then the AHU (ie CCPU) shall determine if it can play chimes through the chime back-up speaker(s). If for some reason the AHU (ie CCPU) cannot play through the back-up speaker then the AHU shall set its AHU_Chimes_Supported signal = Not Supported.

The Chime components are responsible for setting "_Chimes_Supported = Not_Supported" (ex SYNC 4.2 CCPU, DSP AMPv2) whenever they are no longer able to produce chimes because of a fault condition, otherwise their signal shall be set to 'Supported' if able to produce chimes with no fault conditions (see Infotainment Diagnostic spec for details).

- A normal crank event is not considered a fault condition. A normal crank event would not cause the _Chime_Supported signal to change. For example, if Chime_Supported = Supported and a normal crank event

occurs the _Chime_Supported value would remain equal to Supported and would not change because of the crank event (ie would never change to Inactive or Not_Supported).
- A chime FMEA shall be performed on all the chime components to verify that fault conditions are detected and _Chimes_Supported is set to Not_Supported when necessary.

Any infotainment component that could prevent chimes from being played through the infotainment system needs to have a "XXX_Chimes_Supported" signal so chimes can be transferred to the chime back-up speaker when needed for an error condition.

Some fault conditions (while not limited to these) that would result in the DSP AMP variant 2 _Chimes_Supported signal being set to Not_Supported: 1) Short/open circuit to any of the chime speakers 2) short/open circuit on the line level signals to the chime components 3) low voltage preventing the chime components from producing audio (not applicable to crank but if AHU stuck at a low voltage) 4) A2B link error / A2B loss of communication (see A2B SPSS for details)…
- For the cases above if this happens on SYNC 4.2 the SYNC 4.2 CCPU module needs to decide how to handle above:
  - It could decide to switch to the back-up speaker(s) controlled by the CCPU,
  - if CCPU knows there is a redundant speaker with the same directionality that can still play chimes then determine if the back-up speaker should be used or not,
  - If the CCPU cannot play the chime then set AHU_Chime_Supported = Not_Supported so the VMCU can play chime audio through the back-up speaker(s)

When the Audio Enable line is keeping the Audio Muted for more than 5 seconds then the amp being muted shall set its chime signal 'XXX_Chime_Supported' equal to 'Not_Supported' (example DSP AMPv2).

The Chime Audio Components (ex.SYNC 4.2 CCPU, DSP AMPv2…) shall set their _Chime_Supported signal to Supported or Not_Supported within 2000 msec of the start of Chime Diagnostics. Reference the IDS (infotainment diagnostic spec) for chime diagnostics.

If _Chime_Supported = Supported and then Power_Up_Chime_Modules = transitions from Active to Inactive and the module powers down and is no longer capable of producing chimes then the _Chime_Supported signal shall equal “inactive”.
- Note: the start of chime diagnostics requires _Chime_Supported signal to equal Inactive (see IDS for details)

If the Chime Generator (ie CCPU) doesn't receive the DSP AMPv2 DSP_Chimes_Supported signal equal to Supported within 3000 msec after the start of chime diagnostics it shall treat 'Inactive' the same as 'Not Supported'.

If the Chime Client (Cluster) receives 'AHU_Chimes_Supported = Inactive' 5000 msec after Power_Up_Chime_Modules transitions from Inactive to Active (or at network bus start-up might just go directly to active) then it shall treat 'Inactive' the same as 'Not_Supported'.

Note: The IPC_Infotainment : Chime_Source signal has no effect on the _Chime_Supported signal.

A periodic (if applicable) IPC_Chime/IPC_Chime2 network message is not considered a new chime event for the Chime Audio Components (ex Chime Generator, Chime Audio Source).

The user adjusted Bass, Treble, Balance, Mid-Range, Fade settings shall not affect chimes.
- For example if the BTMBF setting did effect chimes the user could fade to front speakers and a chime that is supposed to be played out of the rear speakers only would not be heard.

~~If the Chime Generator / Chime Audio Source (ie CCPU) is NOT configured to support the chime strategy as defined in this Chime SPSS feature/section than the Chime Generator/Chime Audio Source shall always be set to _Chime_Supported = Not_Supported.~~
- ~~Ex. AHU_Chime_Supported would always be set to Not_Supported if the C1MCA chime strategy was configured ON since C1MCA chime strategy does not support this chime strategy as defined in this feature/section.~~
- ~~Reference the Infotainment Diagnostic Spec for configuration used for this chime feature/section.~~

If there are not event-periodic (1000 msec) chime signals (like CAN set-up) between the VMCU and CCPU then new error handling needs to be developed in case of a missing IPC_Chime, IPC_Chime2, IPC_Infotainment and AHU_Alert signal.
- For example (without event-periodic signals, but only event signals),

1. if there is a continuous chime playing, and then:
2. the IPC_Chime : No_Chime is sent to the CCPU to end the chime but the signal is lost (ex long cold crank in cold weather for a few seconds), then if the IPC_Chime signal is no longer periodic then the CCPU could potentially play the chime indefinitely for that ignition cycle.
    - This won't be an issue if the signals are made event-periodic like CAN or if some other error handling strategy is developed.

3.1.4.3 ALERTv2-SR-REQ-372147/A- Chime_Supported signal changes to Supported from Not_Supported - variant 2

If the Chime Generator set its signal 'AHU_Chimes_Supported' equal to 'Supported' after previously being set to 'Not_Supported' in the same ignition cycle, then:

- the Cluster VMCU shall no longer be the back-up chime generator and the CCPU shall be the back-up chime generator.
    - Note: the Cluster VMCU would need to send the IPC_Chime and IPC_Chime2 messages
- The Cluster VMCU shall keep the Power_Up_Chime_Modules = Inactive with the Chime_Source = Cluster for that ignition cycle.

3.1.4.4 ALERTv2-SR-REQ-372081/A-Load Shed - SYNC 4.2

During an infotainment Load Shed event (HMIAudioMode = Load Shed) the chime back-up speaker(s) shall be used and the Cluster VMCU shall set Power_Up_Chime_Source = Inactive and Chime_Source = Cluster.

- VMCU shall be the back-up generator for early chime generator and when the CCPU is powered down
- CCPU shall be the back-up generator when CCPU powered up (example if CCPU remained powered up for an emergency call or some other high priority event).

3.1.4.5 PWRMAN-SR-REQ-014509/F-Infotainment Components Load Shed State requirements (TcSE ROIN-66172-3)

Unless otherwise noted the infotainment components shall transition to their Standby or Sleep Load Shed low power state when the signal HMIAudioMode == Load Shed.

In the Standby Load Shed low power state non-essential component functions shall be turned OFF (ex. active pre-fetch). Basic standby operations will still be followed such as supporting the Network bus and any regulatory requirements.

Note: There may also be applicable Climate Control load shed requirements for modules that support Climate Control functionality. See Climate specifications for details.

Chimes and Load Shed:

- The infotainment components that support chimes (ex. AHU, DSP AMP, AAM, ANC…) during a transition to load shed from state where chimes are through the infotainment system shall wait until the Cluster transfers control of the chimes back to the Cluster (as defined in ALERT-REQ-014761-Load Shed) before entering their low power states.
- Since the infotainment components that support chimes have to wait for Cluster to transfer chime control back to the Cluster (ie IPC_Infotainment : Chime_Source = Cluster) during a load shed event before they no longer support chimes the infotainment components would have Chime_Supported = Supported while supporting chimes.
- After chime control is transferred to the Cluster the infotainment components shall change Chime_Supported = Not_Supported while the load shed is active.

Chimes and Load Shed – variant 2: (applies to SYNC 4.2)

- SYNC 4.2 (AHU and Cluster in one module): when HMIAudioMode = Load Shed the integrated Chime Client and Chime Generator shall use the back-up speaker(s) and set Power_Up_Chime_Modules = Inactive and Chime_Source = Cluster (as defined in "Alertv2-REQ-372081-Load Shed – SYNC 4.2").

3.1.4.6 ALERT-SR-REQ-014762/D-Response when chime signals from Chime Client set to inactive (TcSE ROIN-193435-1)

If any of the signals Chime, Chime_Directionality, Chime_Time_Criticality, OFF_Time_Btwn_Chime, or Chime_Occurance in the IPC_Chime (1 or 2) messages are set to 'inactive' then the Chime Generator won't create a new chime sound (ex Chime_8 DNA B) and the AHU_Alert : AHU_Chime_Active signal will be set to No_Chime to tell the Cluster that it is not playing the chime.

3.1.4.7 ALERT-SR-REQ-376198/A-Transport Mode - SYNC 4.2

During Transport Mode the chime back-up speaker(s) shall be used and the Cluster VMCU shall set Power_Up_Chime_Modules = Inactive and Chime_Source = Cluster.

- VMCU shall be the back-up generator for early chime generator and when the CCPU is down
- CCPU shall be the back-up generator when CCPU powered up.

3.1.4.8 PWRMAN-SR-REQ-014520/G-Transport Mode and CGEA Chimes (TcSE ROIN-40663-3)

Audio Chimes shall NOT be enabled through the Infotainment System during Transport Mode. The Cluster shall support Chimes during Transport Mode.

During a transition to Transport Mode from another LifeCycMde_D_Actl/CarMode state where the chimes are through the infotainment system the Cluster shall set the 'Chime_Source' signal equal to 'Cluster' and Power_Up_Chime_Modules = Inactive.

- The infotainment components that support chimes shall wait until the Cluster transfers control of the chimes back to the Cluster with the 'Chime_Source = Cluster' AND 'Power_Up_Chime_Module = Inactive' before entering their transport mode low power states.
    - Since the infotainment components that support chimes have to wait for Cluster to transfer chime control back to the Cluster during a transport mode event before they no longer support chimes the infotainment components would have Chime_Supported = Supported while supporting chimes. After chime control is transferred to the Cluster the infotainment components can change Chime_Supported = Not_Supported while transport mode is active.

3.1.5 Sequence Diagrams

Note:

1. In the sequence diagrams for SYNC 4.2 the Chime Generator and Chime Audio Source are the same module and does not include the DSP AMP. Because of this the interface between the Chime Generator and Chime Audio Source can be ignored
2. Any sequence diagrams that call out <CAN> should ignore the CAN label and treat the signals as logical signals not tied to specific network
3. Some sequence diagrams (ex on different alert channels in particular) still show the Cluster sending chime requests without waiting for the no chime from the AHU. For SYNC 4.2 this does not apply, and the Cluster shall follow requirement "Alert-REQ-014741-Cluster Chime-Requests" which requires the Cluster to wait for the no chime. The reason the sequence diagrams where left inside this spec is so SYNC can have common chime software with SYNC 4.1 which might be paired with an external legacy cluster that still does this.

3.1.5.1 ALERT-SD-REQ-014763/D-Cluster ending Continuous Chime event, or Fixed Repetition Chime event where Cluster ends chime before repetitions finished (TcSE ROIN-40403-4)

Pre-condition

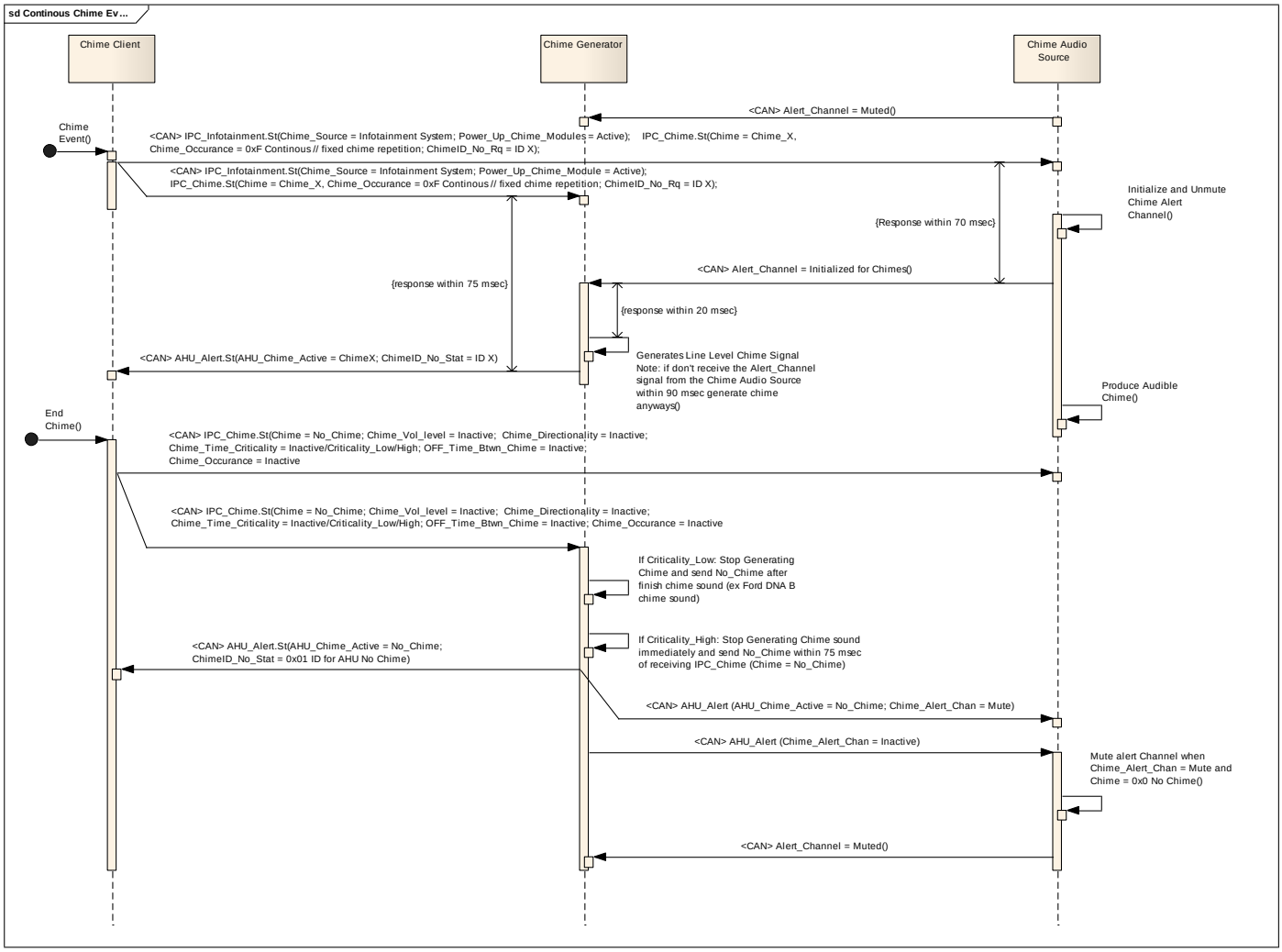
No Chimes are active

Scenario

Chime Client initiates a chime

Post-condition
Chime is ended by the Chime Client

Sequence Diagram



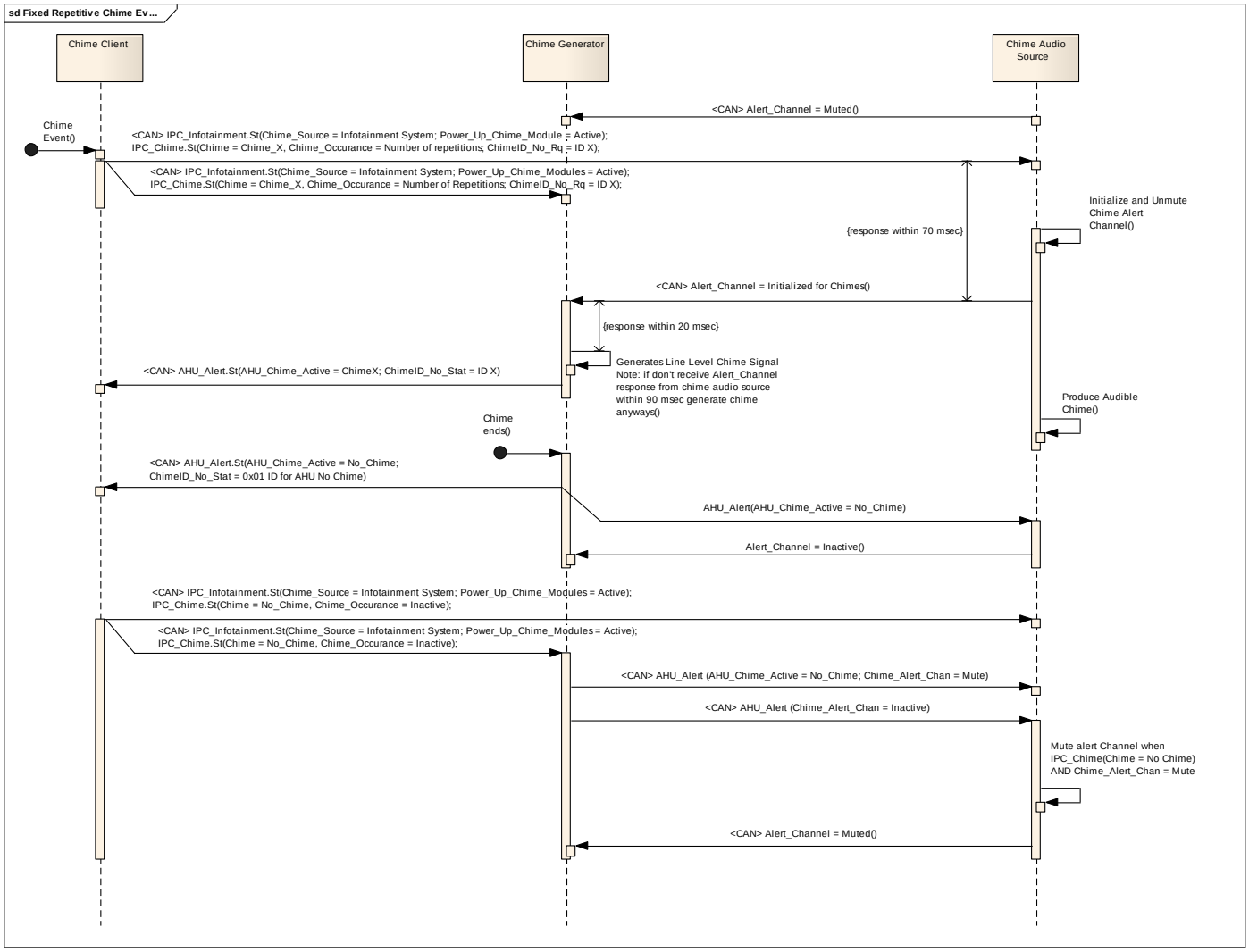
3.1.5.2 ALERT-SD-REQ-014764/B-Fixed Repetitive Chime Event (TcSE ROIN-167429-2)

Pre-condition
No Chimes are active

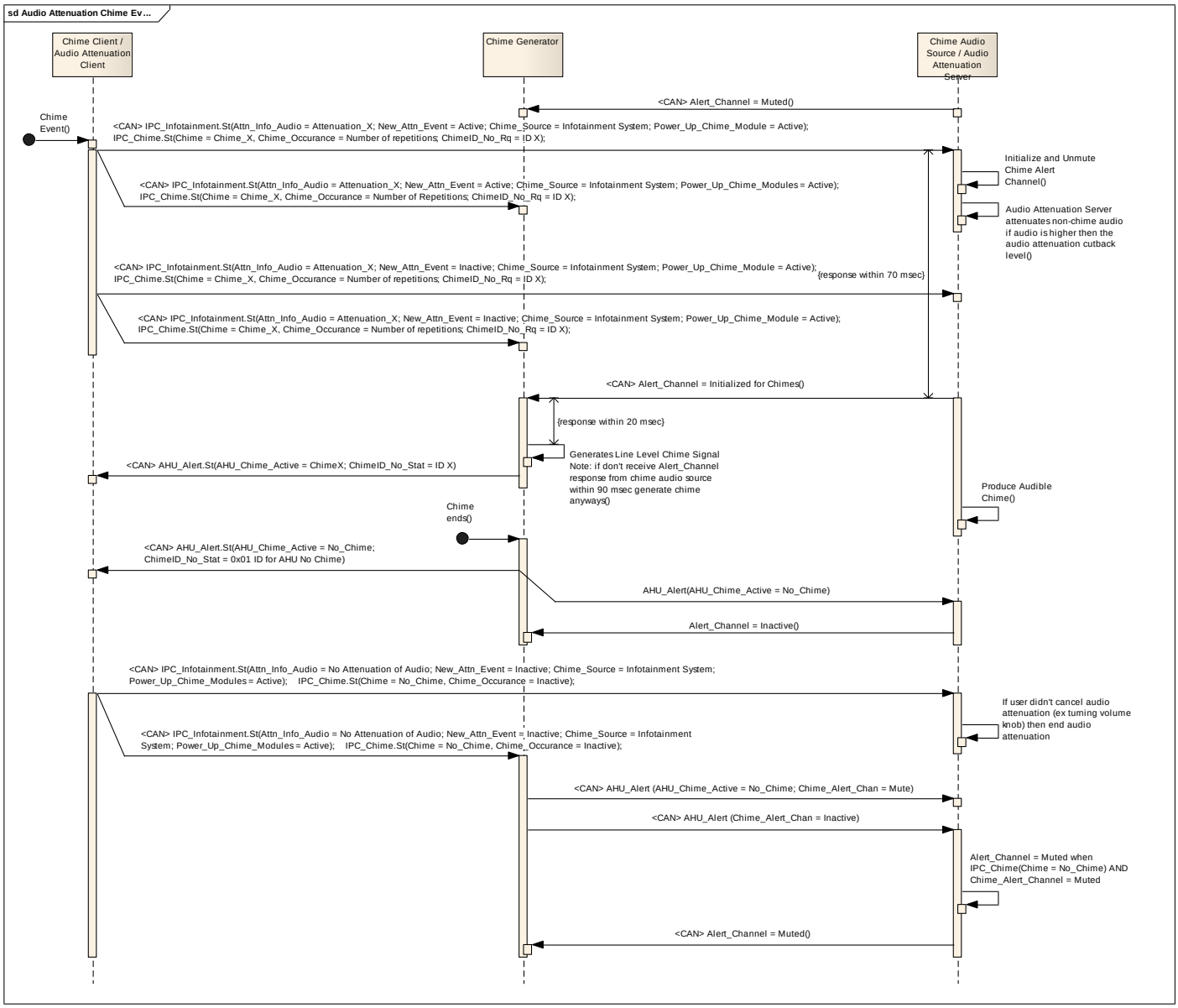
Scenario
Chime Client initiates a Fixed Chime Event

Post-condition
Chime is completed

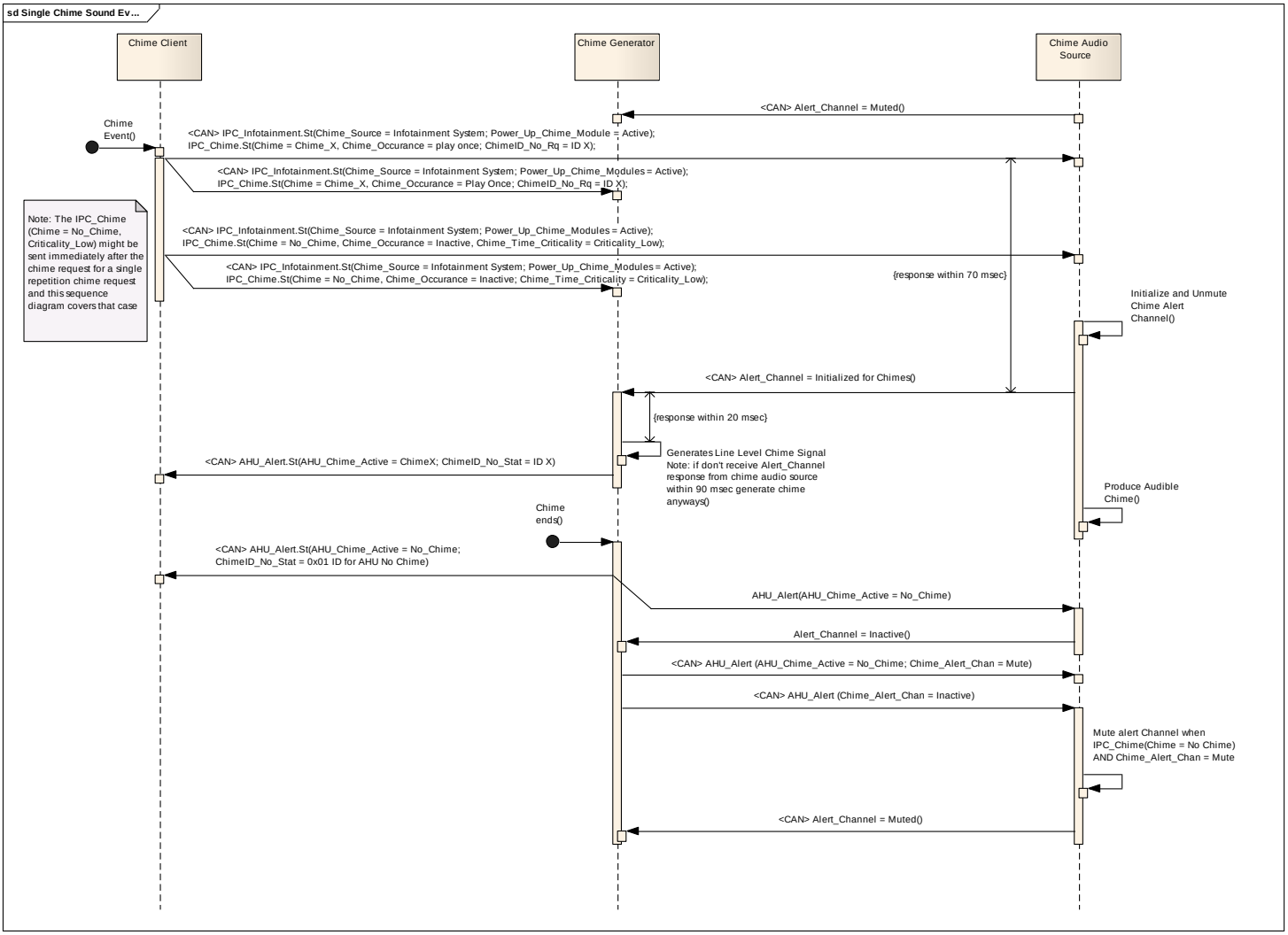
Sequence Diagram



Sequence Diagram


sd Audio Attenuation Chime Ev...
Chime Client / Audio Attenuation Client
Chime Generator
Chime Audio Source / Audio Attenuation Server
<CAN> Alert_Channel = Muted()
Chime Event()
<CAN> IPC_Infotainment.St(Attn_Info_Audio = Attenuation_X; New_Attn_Event = Active; Chime_Source = Infotainment System; Power_Up_Chime_Module = Active); IPC_Chime.St(Chime = Chime_X, Chime_Occurance = Number of repetitions; ChimeID_No_Rq = ID X);
Initialize and Unmute Chime Alert Channel()
<CAN> IPC_Infotainment.St(Attn_Info_Audio = Attenuation_X; New_Attn_Event = Active; Chime_Source = Infotainment System; Power_Up_Chime_Modules = Active); IPC_Chime.St(Chime = Chime_X, Chime_Occurance = Number of Repetitions; ChimeID_No_Rq = ID X);
Audio Attenuation Server attenuates non-chime audio if audio is higher then the audio attenuation cutback level()
<CAN> IPC_Infotainment.St(Attn_Info_Audio = Attenuation_X; New_Attn_Event = Inactive; Chime_Source = Infotainment System; Power_Up_Chime_Module = Active); IPC_Chime.St(Chime = Chime_X, Chime_Occurance = Number of repetitions; ChimeID_No_Rq = ID X);
{response within 70 msec}
<CAN> IPC_Infotainment.St(Attn_Info_Audio = Attenuation_X; New_Attn_Event = Inactive; Chime_Source = Infotainment System; Power_Up_Chime_Module = Active); IPC_Chime.St(Chime = Chime_X, Chime_Occurance = Number of repetitions; ChimeID_No_Rq = ID X);
<CAN> Alert_Channel = Initialized for Chimes()
{response within 20 msec}
Generates Line Level Chime Signal Note: if don't receive Alert_Channel response from chime audio source within 90 msec generate chime anyways()
<CAN> AHU_Alert.St(AHU_Chime_Active = ChimeX; ChimeID_No_Stat = ID X)
Produce Audible Chime()
Chime ends()
<CAN> AHU_Alert.St(AHU_Chime_Active = No_Chime; ChimeID_No_Stat = 0x01 ID for AHU No Chime)
AHU_Alert(AHU_Chime_Active = No_Chime)
Alert_Channel = Inactive()
<CAN> IPC_Infotainment.St(Attn_Info_Audio = No Attenuation of Audio; New_Attn_Event = Inactive; Chime_Source = Infotainment System; Power_Up_Chime_Modules = Active); IPC_Chime.St(Chime = No_Chime, Chime_Occurance = Inactive);
If user didn't cancel audio attenuation (ex turning volume knob) then end audio attenuation
<CAN> IPC_Infotainment.St(Attn_Info_Audio = No Attenuation of Audio; New_Attn_Event = Inactive; Chime_Source = Infotainment System; Power_Up_Chime_Modules = Active); IPC_Chime.St(Chime = No_Chime, Chime_Occurance = Inactive);
<CAN> AHU_Alert (AHU_Chime_Active = No_Chime; Chime_Alert_Chan = Mute)
<CAN> AHU_Alert (Chime_Alert_Chan = Inactive)
Alert_Channel = Muted when IPC_Chime(Chime = No_Chime) AND Chime_Alert_Channel = Muted
<CAN> Alert_Channel = Muted()

Sequence Diagram



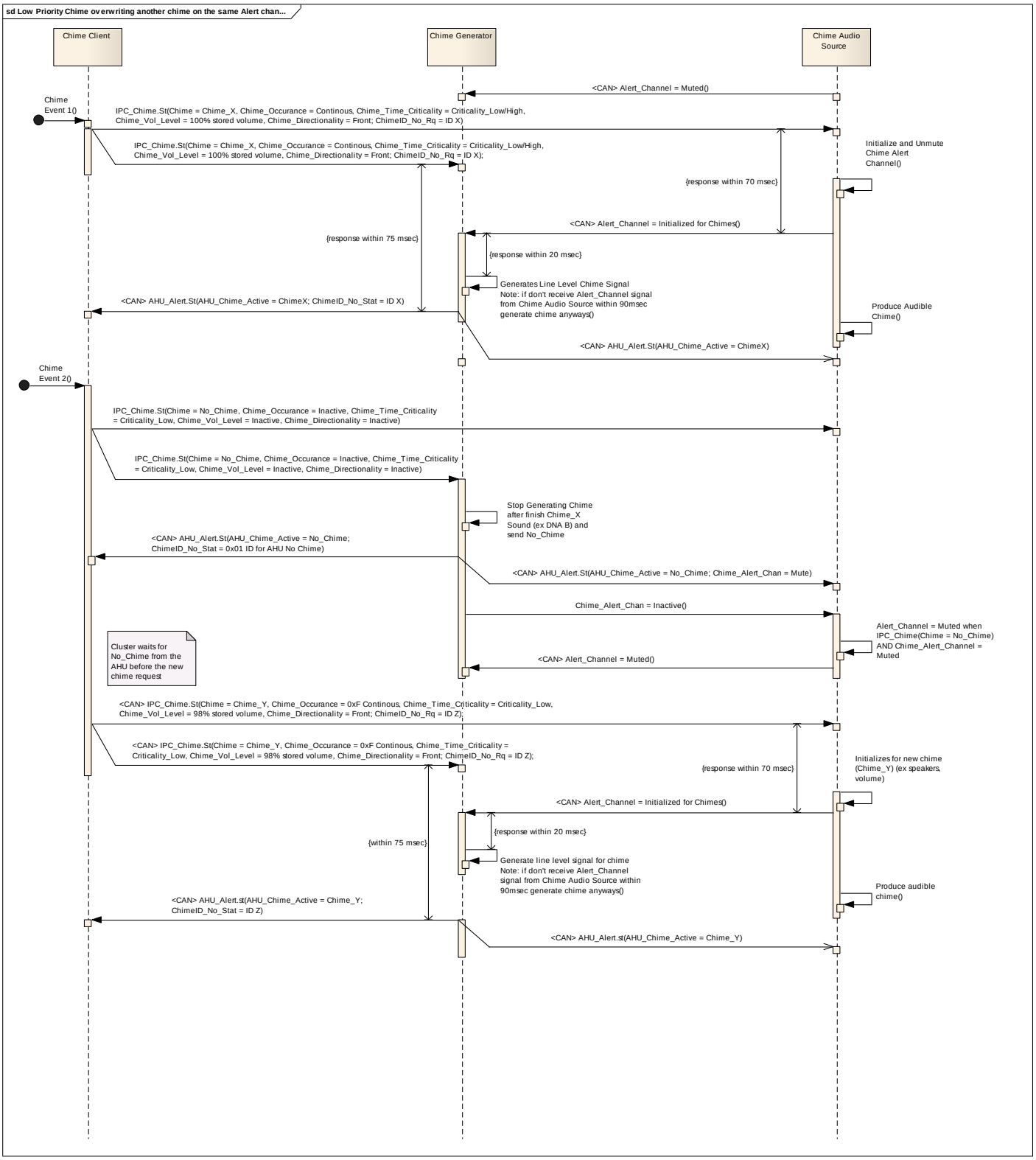
3.1.5.3 ALERT-SD-REQ-014765/A-Low Priority Chime overwriting another Chime on the same Alert Channel (TcSE ROIN-200749-2)

Pre-condition
Chime event 1 is active on Alert Channel 1

Scenario
Chime Client initiates a low criticality Chime event 2 on Alert Channel 1

Post-condition
Chime event 2 is playing through the infotainment system on Alert Channel 1

Sequence Diagram



3.1.5.4 ALERT-SD-REQ-014766/A-High Priority Chime overwriting another Chime on the same Alert Channel (TcSE ROIN-200760-3)

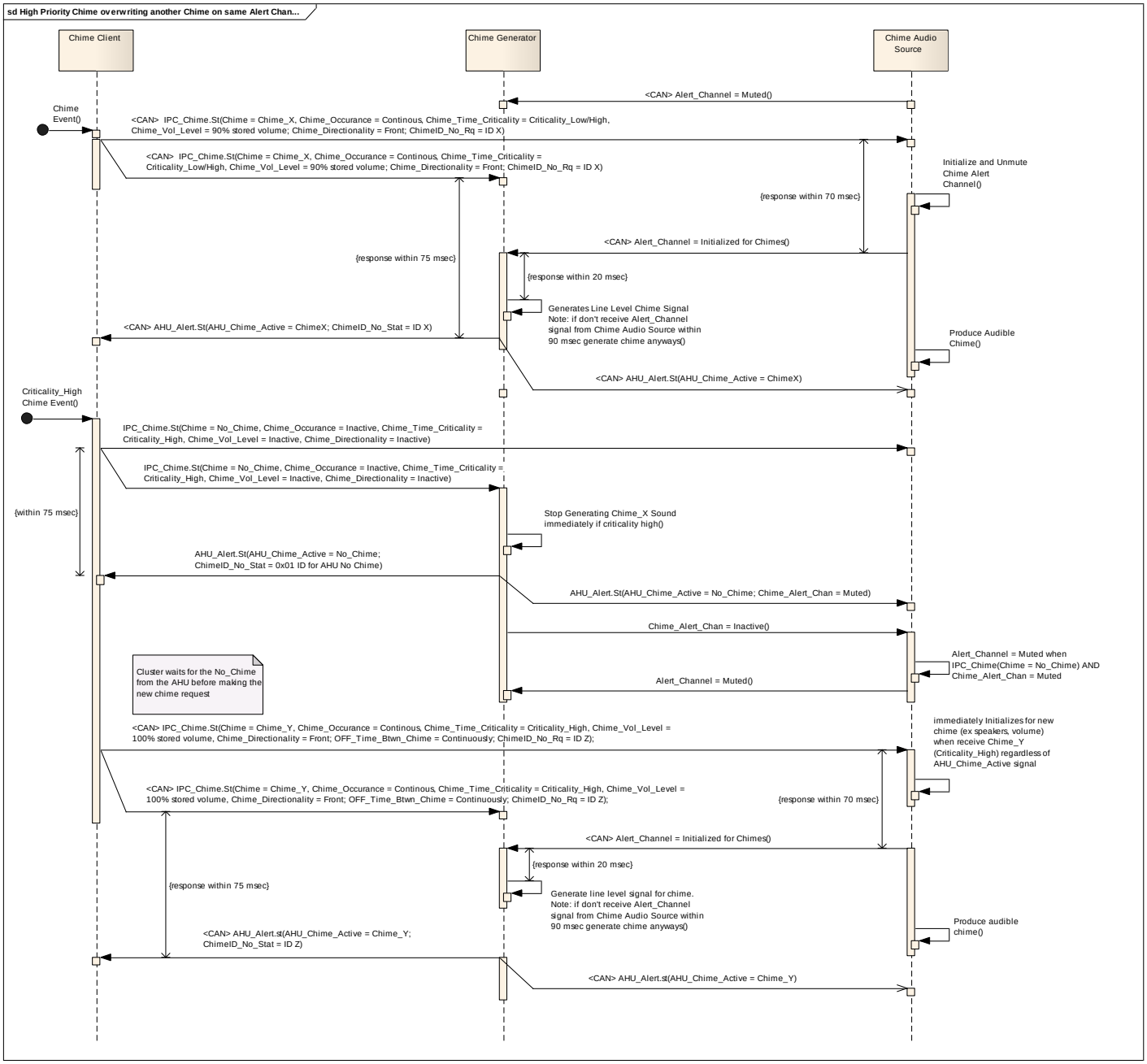
Pre-condition

There is a chime active on Alert Channel 1

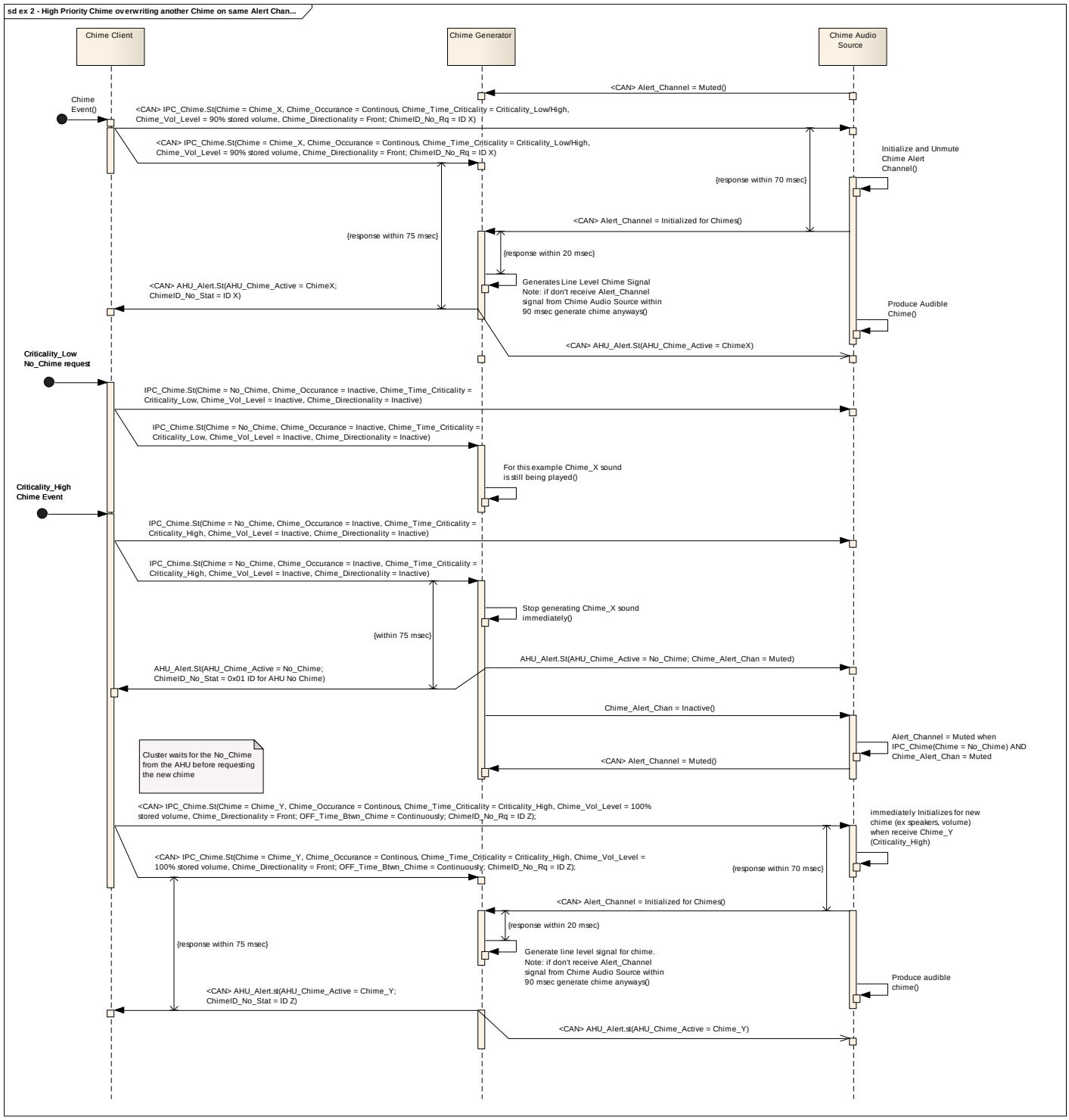
Scenario
Chime Client initiates a High Criticality Chime on Alert Channel 1

Post-condition
The new high priority chime is playing through the infotainment system on Alert Channel 1

Sequence Diagram



Sequence Diagram



3.1.5.5 ALERT-SD-REQ-014767/A-Low Priority Chime - requesting a new chime on a different Alert channel with the same directionality (TcSE ROIN-285762-1)

Pre-condition

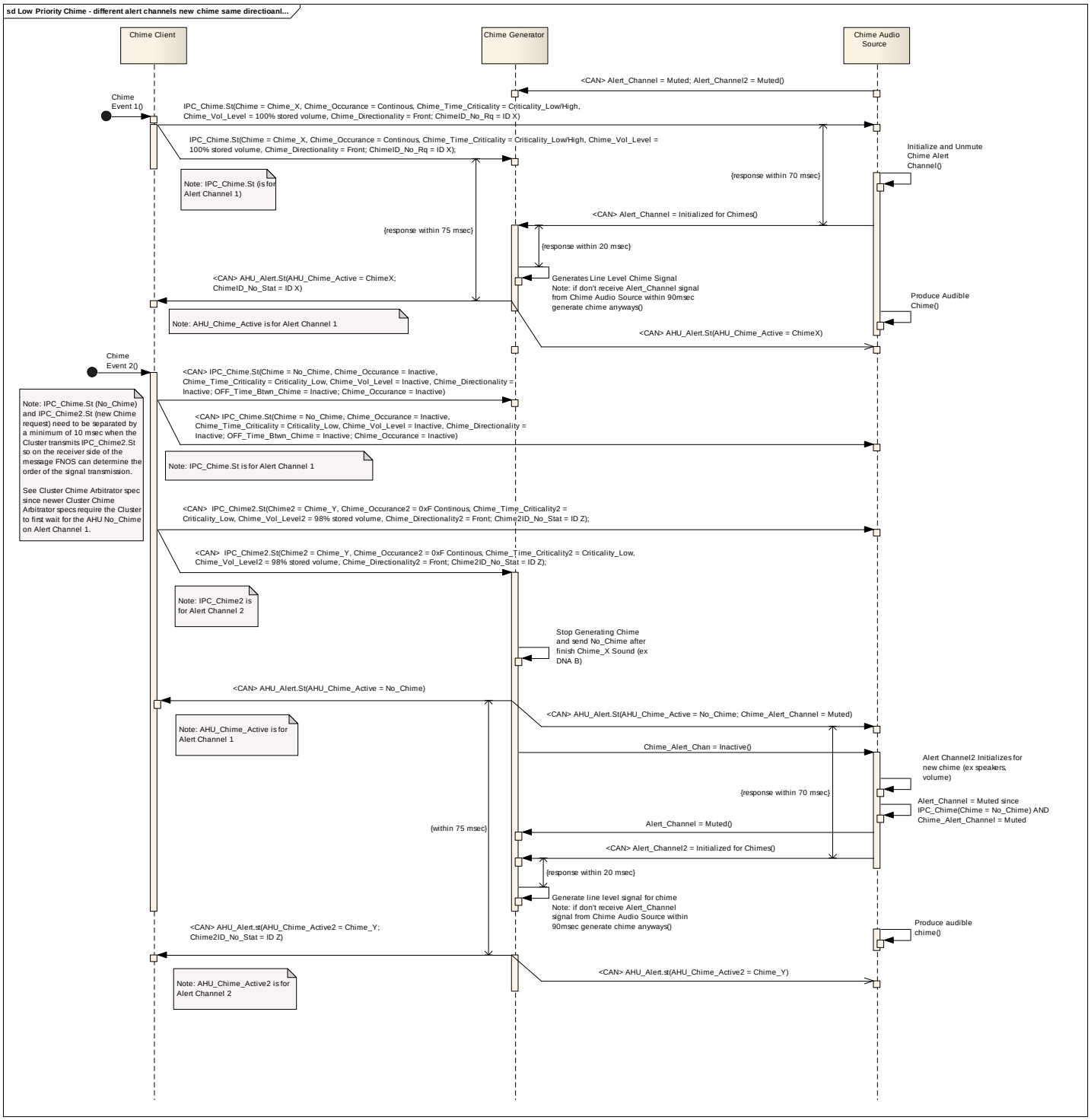
Chime event 1 is active on Alert Channel 1

Scenario

Chime Client initiates a new low criticality Chime event 2 on Alert Channel 2

Post-condition
Chime event 2 is playing through the infotainment system on Alert Channel 2

Sequence Diagram



3.1.5.6 ALERT-SD-REQ-014768/D-High Priority Chime - requesting a new chime on a different Alert channel with the same directionality (TcSE ROIN-285767-1)

Pre-condition

Chime event 1 is active on Alert Channel 1

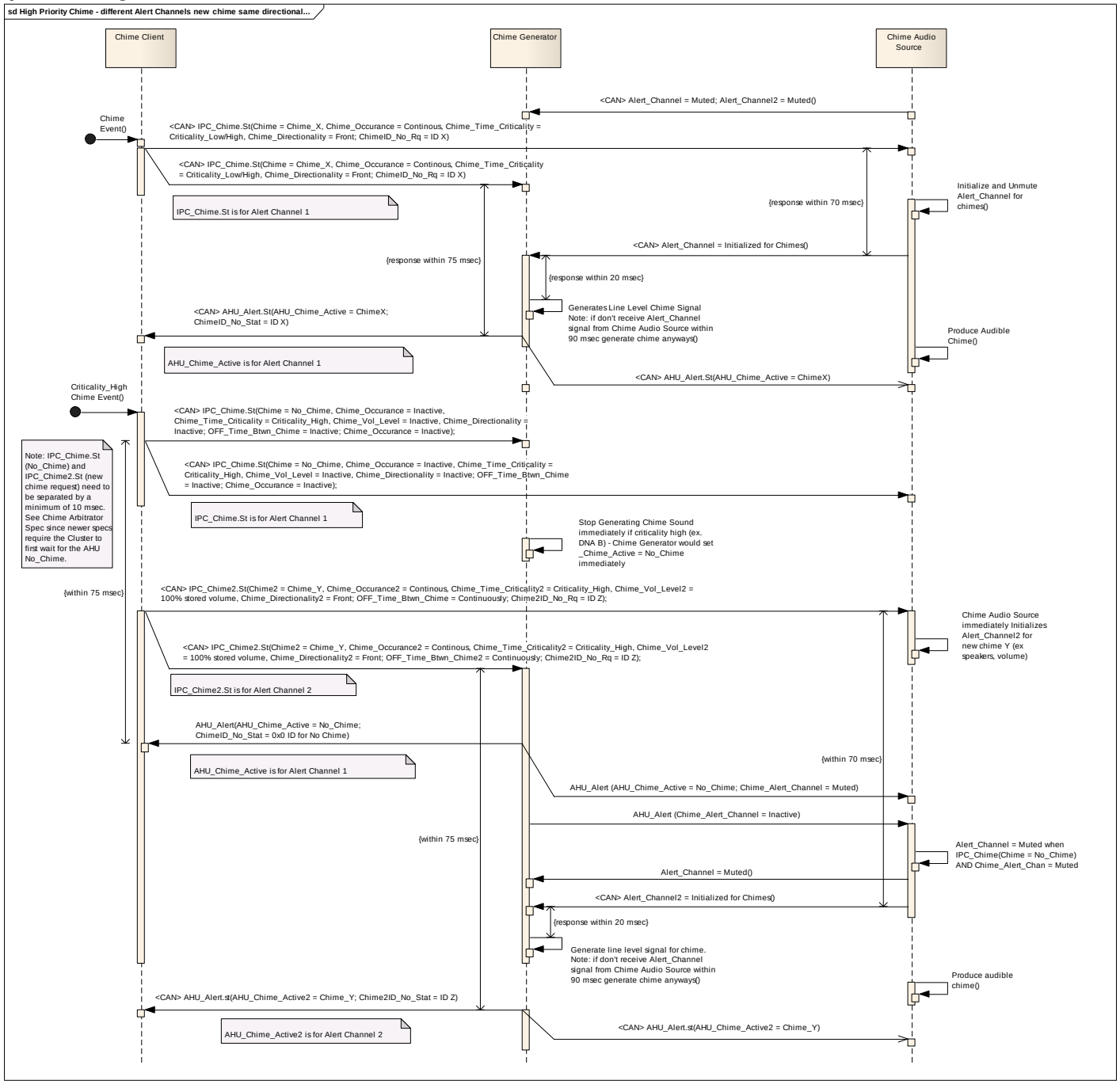
Scenario

Chime Client initiates a new high criticality Chime event 2 on Alert Channel 2

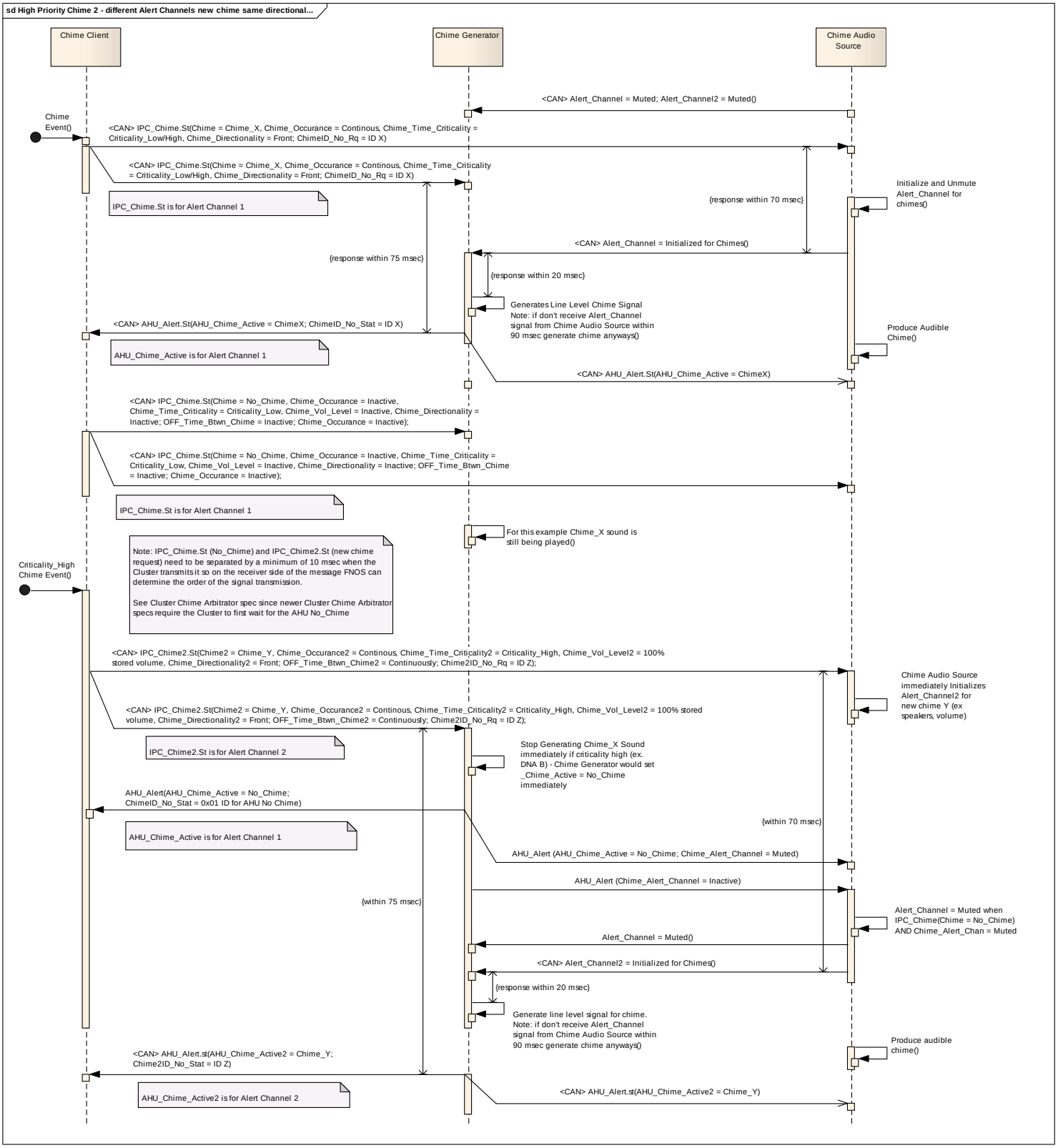
Post-condition

Chime event 2 is playing through the infotainment system on Alert Channel 2

Sequence Diagram



Sequence Diagram



3.2 ALERT-FUN-REQ-195348/C-Adjustable Chime Strategy

3.2.1 Use Cases

3.2.1.1 ALERT-UC-REQ-195369/D-Set rear park aid chime volume level while no other chime is active

Actors	Vehicle Occupant
Pre-conditions	Infotainment System is Powered ON The HMI is configured to have Rear Park Aid chime volume adjustment HMI available when Rear Park Aid is on the vehicle. Park Aid chime adjustment HMI is the active HMI screen No chime is active.
Scenario Description	The user changes the chime volume level of the rear park aid chime via the HMI
Post-conditions	The rear park aid sound/feedback chime is played at the new volume level. If infotainment audio was playing (ex AM/FM/USB…) then the audio is attenuated to the attenuation level that is normally used when playing the rear park aid chime. The new volume level of the rear parking aid chime is changed and stored (stored between ignition cycles). The new rear park aid chime volume level is shown on the HMI.
Notes	This use case was for rear park aid chime but applies to any supported adjustable chime. The attenuation signal only applies to chimes that support chime attenuation. The HMI for adjustable chime settings should only be shown when ignition is in Run
Interfaces	Vehicle System, G-HMI

3.2.1.2 ALERT-UC-REQ-195370/D-Set front park aid chime volume level while no other chime is active

Actors	Vehicle Occupant
Pre-conditions	Infotainment System is Powered ON. The HMI is configured to have Front Park Aid chime volume adjustment HMI available when Front Park Aid is on the vehicle Park Aid chime adjustment HMI is the active HMI screen. No chime is active.
Scenario Description	The user changes the chime volume level of the front park aid chime via the HMI
Post-conditions	The front park aid sound/feedback chime is played with the new volume level. If infotainment audio was playing (ex AM/FM/USB…) then the audio is attenuated to the attenuation level that is normally used when playing the front park aid chime.

	The new volume level of the front parking aid chime is changed and stored (stored between ignition cycles). The new front park aid chime volume level is shown on the HMI.
Notes	This use case was for front park aid chime but applies to any supported adjustable chime. The attenuation signal only applies to chimes that support chime attenuation. The HMI for adjustable chime settings should only be shown when ignition is in Run
Interfaces	Vehicle System, G-HMI

3.2.1.3 ALERT-UC-REQ-195371/E-Set front/rear park aid chime volume level while another chime is playing through the same speakers (Front or Rear speakers)

Actors	Vehicle Occupant
Pre-conditions	Infotainment System is Powered ON. The HMI is configured to have Front Park Aid chime volume adjustment HMI available when Front Park Aid is on the vehicle Park Aid chime adjustment HMI is the active HMI screen. A non front park aid chime is currently playing out of the front speakers (ex low fuel warning chime, seat belt chime…) through the infotainment system
Scenario Description	The user changes the chime volume level of the front park aid chime via the HMI
Post-conditions	If there is a currently active chime (ex seat belt chime) playing out of the same speakers as the front park aid chime (ie directionality equals front speakers) then the front park aid sound/feedback chime is not played. If there is a currently active chime (ex seat belt chime) playing out of the same speakers as the front park aid chime (ie directionality equals front speakers) then the audio attenuation associated with the front park aid chime is not sent (audio attenuation could be sent for the active chime though if that chime has audio attenuation associated with it). The new volume level of the front parking aid chime is changed and stored (stored between ignition cycles). This happens whether or not the front park aid feedback chime was played. The new front park aid chime volume level is shown on the HMI.
Notes	This use case applies to any supported adjustable chime and not just front park aid chime The HMI for adjustable chime settings should only be shown when ignition is in Run
Interfaces	Vehicle System, G-HMI

3.2.1.4 ALERT-UC-REQ-195372/F-Set front/rear park aid chime volume level while the front/rear park aid chime is already playing

Actors	Vehicle Occupant
Pre-conditions	Infotainment System is Powered ON. The HMI is configured to have Front Park Aid chime volume adjustment HMI available when Front Park Aid is on the vehicle.

The information contained in this document is Proprietary to Ford Motor Company.

	Park Aid chime adjustment HMI is the active HMI screen. The Front Park Aid chime is currently playing out of the front speakers through the infotainment system
Scenario Description	The user changes the chime volume level of the front park aid chime via the HMI while the front park aid chime is already playing
Post-conditions	The front park aid chime that is already playing is played with the new volume level. This does not change the chime itself other than the volume. The front park aid duration and pause between chimes is not changed with the front park aid volume adjustment. The new volume level of the front parking aid chime is changed and stored. The new front park aid chime volume level is shown on the HMI.
Notes	This use case applies to any supported adjustable chime and not just front park aid chime The HMI for adjustable chime settings should only be shown when ignition is in Run
Interfaces	Vehicle System, G-HMI

3.2.1.5 ALERT-UC-REQ-209067/E-Adjust chime volume while adjustable chime is playing feedback chimes

Actors	Vehicle Occupant
Pre-conditions	Infotainment System is Powered ON. The HMI is configured to have Front Park Aid chime volume adjustment HMI available when Front Park Aid is on the vehicle Park Aid chime adjustment HMI is the active HMI screen. No Chimes are playing through the infotainment system
Scenario Description	1. The user changes (this case increases) the chime volume level of the front park aid chime via the HMI to +1 2. While the feedback chime is playing the user increases the chime volume level again to +2
Post-conditions	The current Chime_16 Forward Park Aid feedback chime sound completes it sound at +1 adjustable chime volume (doesn't play other Chime_16 repetitions for feedback chime at +1) The Chime_16 Forward Park Aid feedback chime is played at the desired number of repetitions at the +2 adjustable chime volume
Notes	This use case applies to reverse park aid also and not just front park aid chime See the Cluster Chime Arbitrator spec for this use case and in case of any conflict the chime arbitrator spec shall take precedent The HMI for adjustable chime settings should only be shown when ignition is in Run
Interfaces	Vehicle System, G-HMI

The information contained in this document is Proprietary to Ford Motor Company.

3.2.1.6 ALERT-UC-REQ-233120/C-Chime event using an Adjustable Chime setting

Actors	Vehicle Occupant
Pre-conditions	Infotainment System is Powered ON. Front Park Aid adjustable chime volume is already set to + 1 adjustable chime value (user previously adjusted the settings in earlier ignition cycles) No Chimes are playing through the infotainment system
Scenario Description	Front Park Aid chime event happens • The Cluster request Front Park Aid chime at the +1 adjustable chime value
Post-conditions	Infotainment System plays the Front Park Aid chime at the +1 adjustable chime loudness level
Notes	This use case applies to any supported adjustable chime and not just front park aid chime The HMI for adjustable chime settings should only be shown when ignition is in Run
Interfaces	Vehicle System

3.2.2 Requirements

For adjustable chimes the Chime ID signal requirement "Alert-SR-REQ-237862-Chime ID signal" has to be supported. Once supported in the AHU software then the Chime ID signal shall be supported by the AHU regardless whether adjustable chimes is supported or not.

3.2.2.1 ALERT-SR-REQ-232946/D-Feature Based Message Protocol usage for adjustable chimes HMI Settings

The definition of how to use Feature Based Message Protocol for Set and Query operations is defined in the Feature Based Message Protocol SPSS and shall be followed to support this SPSS.

To support this SPSS for Adjustable Chimes the supplier will need an August 2016 or later version of the FBMP SPSS to accompany this spec.

~~For Adjustable Chimes the Adjustable Chime Client (ex SYNC) shall use the infotainment start-up method (ie when HMIAudioMode goes from OFF to ON) to query the adjustable chimes setting values from the Chime Client/Adjustable Chime Server (ex Cluster). This is described in the FBMP SPSS.~~

For Adjustable Chimes the Adjustable Chime Client (ex SYNC) shall use the start-up method where the Adjustable Chime Client queries the adjustable chimes setting values from the Chime Client/Adjustable Chime Server (ex Cluster) after ignition goes to Run. This is described in the FBMP SPSS.

3.2.2.2 ALERT-SR-REQ-209035/E-Saving Adjustable Chime Settings between power modes

The Cluster Chime Client / Cluster Adjustable Chime Server shall remember the adjustable chime volume selection values (ex frontParkAidChimveVolSelection.St, rearParkAidChimeVolSelection.St) between power modes. This includes but is not limited to ignition cycles and between network bus wake-up and sleep cycles.

3.2.2.3 ALERT-REQ-208870/E-Adjustable chimes when the Cluster is the chime audio source when chimes are not supported by the infotainment system

When chimes are not through the infotainment system and the cluster back up chime generator is the Chime Audio Source (ex AHU_Chime_Supported = Not_Supported) then the chimes wil be played by the Cluster at the Cluster default chime volume and will not support adjustable chime volume levels.

3.2.2.4 ALERT-HMI-REQ-208871/F-HMI for adjustable chimes when chimes is not through the infotainment system

When AHU_Chime_Supported = Not Supported the Adjustable Chime HMI (ex APIM) shall:

- not allow the user to adjust the chime volume from the HMI adjustable chime menu.
  - HMI may want to indicate to the user that the adjustable chime feature is not supported (such as greying out the menu pick). HMI team to determine
- Show the adjustable chime volume at the 0 / default volume

The Adjustable Chime HMI Client (ex APIM) shall monitor the AHU_Chime_Supported signal to support this requirement.

Note: Chimes are not through the infotainment system whenever the signal AHU_Chime_Supported = Not Supported. When chimes are not supported by the infotainment system (ex speaker open circuit) then the Cluster back-up chime generator is the Chime Audio Source for that ignition cycle (or until the issue is fixed).

3.2.2.5 ALERT-SR-REQ-208872/G-HMI when adjusting the chime volume

When the Adjustable Chime HMI Client sends a request to change the chime volume to the Cluster Chime Client / Adjustable Chime Server, the HMI shall not show the updated chime volume until the Adjustable Chime HMI Client receives the response from the cluster chime client / Adjustable Chime Server that it is at the new chime level.

Example:

Pre-Condition:
The Adjustable HMI client has displayed the Front Park Aid adjustable chime volume as 0 / default volume.

Event:
The user increases the Front Park Aid adjustable chime volume to +1 from the HMI and the Adjustable Chime HMI Client sends Feature.Rq(Operation = Set; FeatureID = FrontParkAidChimeVolSelection; Configuration = +1 Adjustable Chime Volume Level; PersonalityIndex = Active Personality) to the Cluster Chime Client / Adjustable Chime Server.

The Adjustable HMI Clients HMI still shows the Front Park Aid adjustable chime volume as 0 / default volume.

Post-Condition:
The Adjustable Chime HMI Client (ex SYNC) receives Feature.St(FeatureID = FrontParkAidChimeVolSelection; Configuration = +1 Adjustable Chime Volume Level; PersonalityIndex = Active Personality) and updates the HMI to show Front Park Aid is at +1 adjustable chime volume level.

3.2.2.6 ALERT-REQ-208983/E-Adjustable Chime Client HMI configuration

The Adjustable Chime Client (ex APIM) shall have a configuration for what adjustable chimes are supported or not on the HMI. When adjustable chimes are not supported the HMI for adjustable chimes shall not be shown.

3.2.2.7 ALERT-SR-REQ-209036/L-Mapping Adjustable Chime HMI selection to an Adjustable Chime Sound

The infotainment system chime audio source (ex AHU and DSP AMP) shall use the AdjustableChimeVol.St and AdjustableChimeVol2.St signals in the IPC_Chime / IPC_Chime2 message to map the adjustable chime value to their volume loudness value specified in the module hardware specification / interior harmony specification for a chime sound.

The chime audio source (ex AHU / DSP AMP) shall support adjustable chimes for the following chime sounds in the IPC_Chime / IPC_Chime2 message when Chime / Chime2 equals:

- Chime_16 Forward Park Chime
- Chime_17 Reverse Park Chime
- Chime_27 RPA Continuous
- Chime_28 FPA Continuous

Note: for a complete list of chime sounds see the Chime_X ID assignment list see "Alert-REQ-014755-Chime ID Assignments" in the Chime SPSS.

Selected adjustable chime volume level on the HMI sent to the Cluster adjustable chime server	AdjustableChimeVol.St / AdjustableChimeVol2.St signals value in the IPC_Chime and IPC_Chime2 messages	Mapping loudness value for the Chime Audio Source
-3	-3	See Module Hardware Spec / interior harmony specification for loudness target
-2	-2	See Module Hardware Spec / interior harmony specification for loudness target
-1	-1	See Module Hardware Spec / interior harmony specification for loudness target
0 / default volume	0 / default volume	See Module Hardware Spec / interior harmony specification for loudness target
+1	+1	See Module Hardware Spec / interior harmony specification for loudness target
+2	+2	See Module Hardware Spec / interior harmony specification for loudness target
+3	+3	See Module Hardware Spec / interior harmony specification for loudness target

Note: the AHU and DSP AMP shall have stored the Chime_X adjustable chime loudness values (ie -3, -2, -1, +1, +2, +3) above and below the 100% max loudness value (0 / default volume) for an adjustable Chime_X sound.

The Chime_Vol_Level signal in the IPC_Chime message shall adjust the Chime_X volume levels for both the default level (0) and for the adjustable chime values (ie -3, -2, -1, +1, +2, +3) by the percentage indicated in the Chime_Vol_Level signal.

Note: for the Adjustable HMI Chime Client (ex APIM) and Adjustable Chime Server (ie Cluster) see the Feature Based Message Protocol "AdjustableChimeVolumeSelection – FBMP" signal interface descriptions in the beginning of this SPSS for what adjustable chime HMI interfaces are supported.

3.2.2.8 ALERT-SR-REQ-209680/G-Setting Adjustable Chime signal in the IPC_Chime and IPC_Chime2 chime request messages

When the Cluster Chime Arbitrator Client sends IPC_Chime / IPC_Chime2 infotainment chime requests to the infotainment system to play chimes at an adjustable chime volume loudness level then the "IPC_Chime : AdjustableChimeVol" signal value shall be set and held at the adjustable chime value for the duration of the Chime Request with the other IPC_Chime signal parameters (ie held until No_Chime is sent by the Chime Client).

When the Cluster Chime Client sends a No_Chime the Cluster AdjustableChimeVol signal shall be set to "Inactive".

- In other words the Cluster will send "IPC_Chime (Chime = No_Chime, AdjustableChimeVol = Inactive").

Note: this could be for the Cluster requesting a standard chime at an adjustable chime level or requesting a feedback chime when the user adjusts the chime volume level.

3.2.2.9 ALERT-SR-REQ-209686/E-Setting Adjustable Chime signal in the AHU_Alert and DSP_AMP_Alert messages

If the AHU or DSP AMP Chime Audio Source gets an IPC_Chime / IPC_Chime2 request to play an adjustable chime for a Chime_X sound that the chime audio source does not support adjustable chimes for then the chime audio source shall set AdjustableChimeVolSupported = Not Supported.

When the AHU or DSP AMP chime audio source plays a chime at an adjustable chime volume loudness level the AdjustableChimeVolSupported signal shall be set for the duration of playing the requested IPC_Chime / IPC_Chime2 chime.

When a DSP AMP is present the AHU shall monitor the DSP_AMP_Alert message and when the DSP AMP responds with "Alert_Chan = Initialized for chimes" the AHU shall:

- set "AHU_Alert : AdjustableChimeVolSupported = Supported" if the DSP AMP sets "AdjustableChimeVolSupported = Supported", else
- set "AHU_Alert : AdjustableChimeVolSupported = Not Supported" if the DSP AMP sets "AdjustableChimeVolSupported = Not Supported"
- continue to keep "AHU_Alert : AdjustableChimeVolSupported = Inactive" if the DSP AMP incorrectly keeps set "AdjustableChimeVolSupported = Inactive"

When the adjustable chime is no longer being played the AdjustableChimeVolSupported signal shall be set back to Inactive

- AHU: (AHU_Alert : AHU_Chime_Active = No_Chime; AdjustableChimeVolSupported = Inactive).
- DSP AMP: (DSP_AMP_Alert : Alert_Channel = Inactive; AdjustableChimeVolSupported = Inactive)

The Cluster only monitors the AHU AdjustableChimeVolSupported signal and does not look at the DSP AMP signal.

See sequence diagrams for examples

3.2.2.10 ALERT-SR-REQ-209764/G-Error Handling when infotainment system doesn't support adjustable chimes for a particular chime sound

If the Chime/Chime2 chime_x sound signals in the IPC_Chime/IPC_Chime2 messages has the AdjustableChimeVol signal set to a value other than "0 adjustable chime volume / default volume" for a chime_X sound that the Chime Audio Source (AHU/DSP AMP) does not support adjustable chimes for then:

1. The Chime Audio Source shall ignore the "IPC_Chime / IPC_Chime2 : AdjustableChimeVol.St / AdjustableChimeVol2.St" signal for the adjustable chime sound it does not support and play the chime through the infotainment system as it normally would as indicated in the other parameters in the IPC_Chime / IPC_Chime2 messages (play the default volume).
    a. The Chime Audio Source would play the chime_x sounds as though AdjustableChimeVol = 0 adjustable chime volume / default volume.
    b. The Chime Audio Source shall not set Chime_Not_Recognized which would cause the Cluster to play the chime. As long as the Infotainment Chime Audio Source supports the chime sound it shall be played by the infotainment system.
2. The Chime Audio Source (ex AHU, DSP AMP) shall set AdjustableChimeSupported or AdjustableChimeSupported2 = Not Supported for the chime sound (ex Chime_X in "Alert-014755_Chime ID Assignments" table) that it does not support adjustable chimes for while it is playing the requested chime.
3. If the DSP AMP is the Chime Audio Source and if the DSP AMP AdjustableChimeVol / AdjustableChimeVol2 signal equals "Not Supported" while "DSP_AMP_Alert : Alert_Chan = Initialized for Chimes" then the AHU shall set:
    a. AHU_Alert (AHU_Chime_Active / AHU_Chime_Active2 = Chime_X) for the chime it is playing through the infotainment system for the chime sound the DSP AMP doesn't support, and
    b. At the same time set AHU_Alert (AdjustableChimeVolSupported / AjustableChimeVolSupported2) to Not_Supported.
4. When the Cluster Chime Client receives AHU_Alert : AdjustableChimeSupported / AdjustableChimeSupported2 = Not_Supported then the Cluster Chime Client shall:
    a. set 0 / Default Chime in the Feature.St message so the HMI is updated to show that the chime volume is not staying at the adjustable chime volume level selected by the user but at the default chime volume.
        i. Note: Cluster would have to update the FBMP Feature.St(FeatureID = Applicable Adjustable Chime; Configuration = 0 / default adjustable chime volume) to update the Adjustable Chime Client HMI to show that still at the default chime volume.

Note: See sequence diagram
"Alert-SD-209691-User tries to change adjustable chime value for a chime sound the infotainment system doesn't support adjustable chimes for" for details of the order the signals are sent.

3.2.3 Sequence Diagrams

3.2.3.1 ALERT-SD-REQ-209690/G-User sets an adjustable chime loudness value for a chime

Pre-condition:

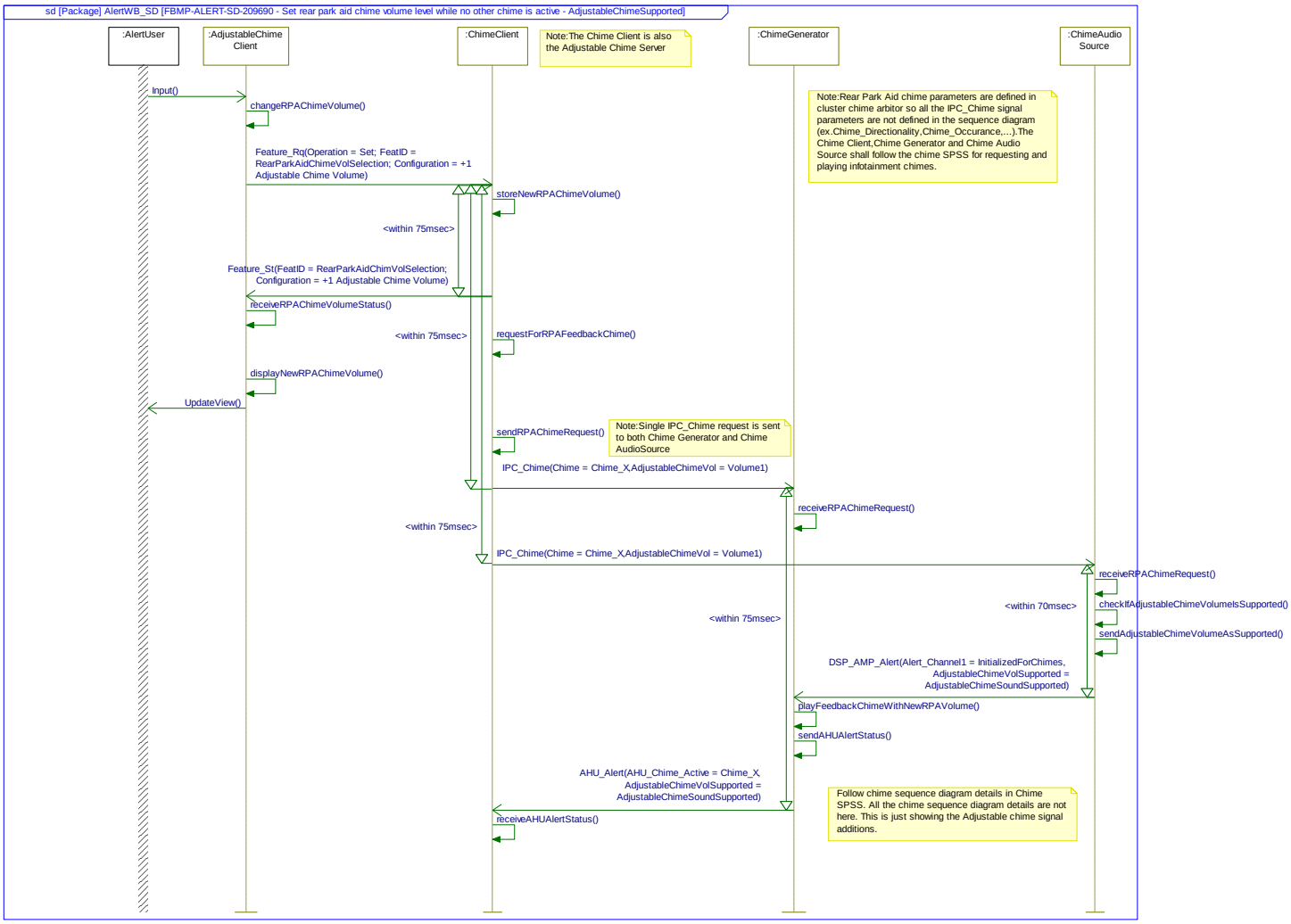
Adjustable chimes HMI screen is shown
Adjustable chimes is set to 0 / default volume loudness value for Reverse Park Aid (note: says RPA as an example but applies for any adjustable chime)
No chime is active

Event:

The user changes the reverse park aid chime loudness value from 0 / default to +1

Post-Condition:

The RPA feedback Chime is played at the +1 adjustable chime value
The HMI shows RPA chime at the +1 adjustable chime value



3.2.3.2 ALERT-SD-REQ-209691/F-User tries to change adjustable chime value for a chime sound the infotainment system doesn't support

Pre-condition:

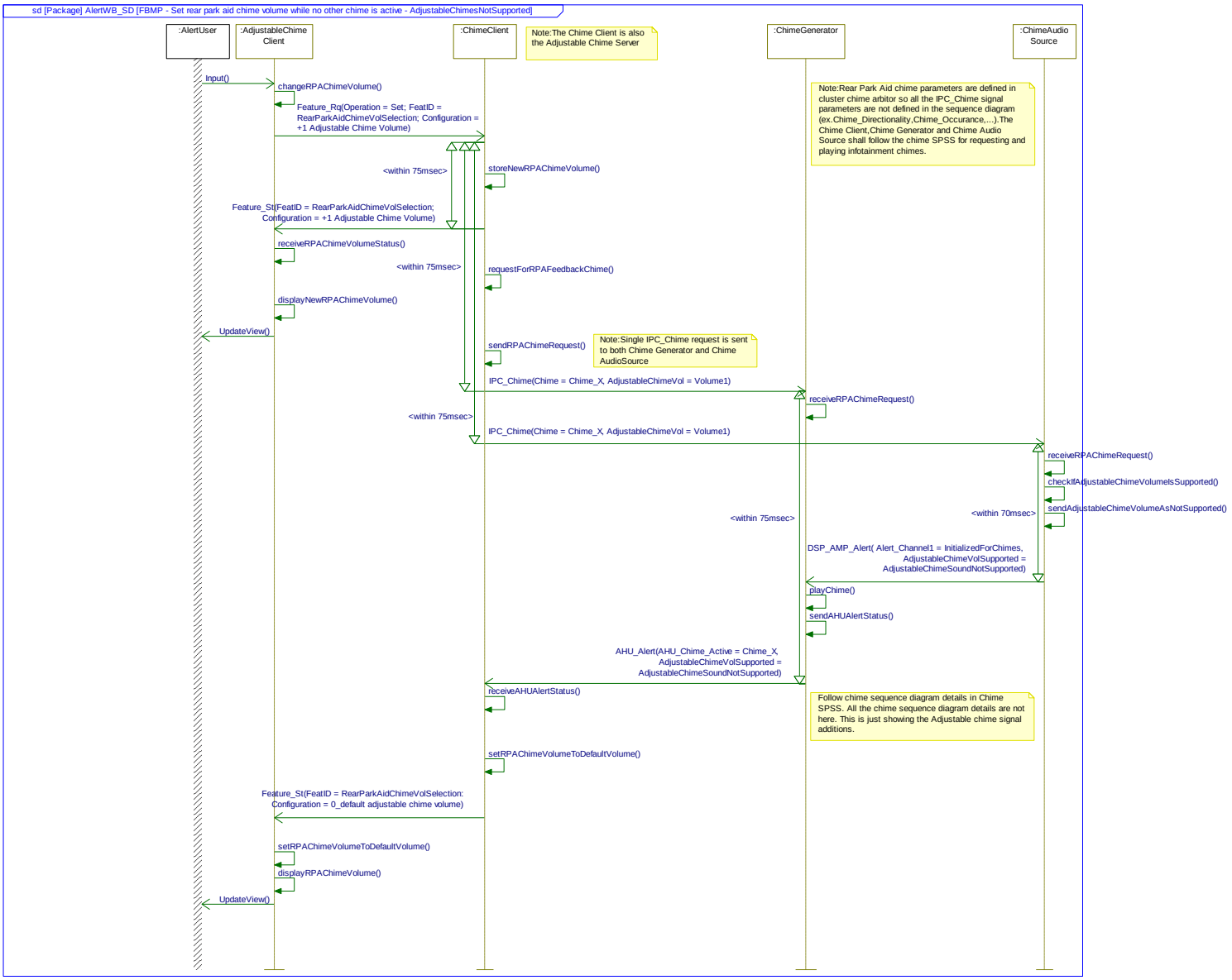
Adjustable Chimes HMI screen is shown
Adjustable chimes is set to 0 / default volume loudness value for Reverse Park Aid (note: says RPA as an example but applies for any adjustable chime).
No Chime is active

Event:

The user changes the reverse park aid chime loudness value from 0 / default to +1

Post-Condition:

RPA feedback chime is played at the default volume
HMI is shown at the RPA default volume



3.2.3.3 ALERT-SD-REQ-233176/B-Normal Chime Event using the adjustable chime signal

Pre-condition:

No Chime is active

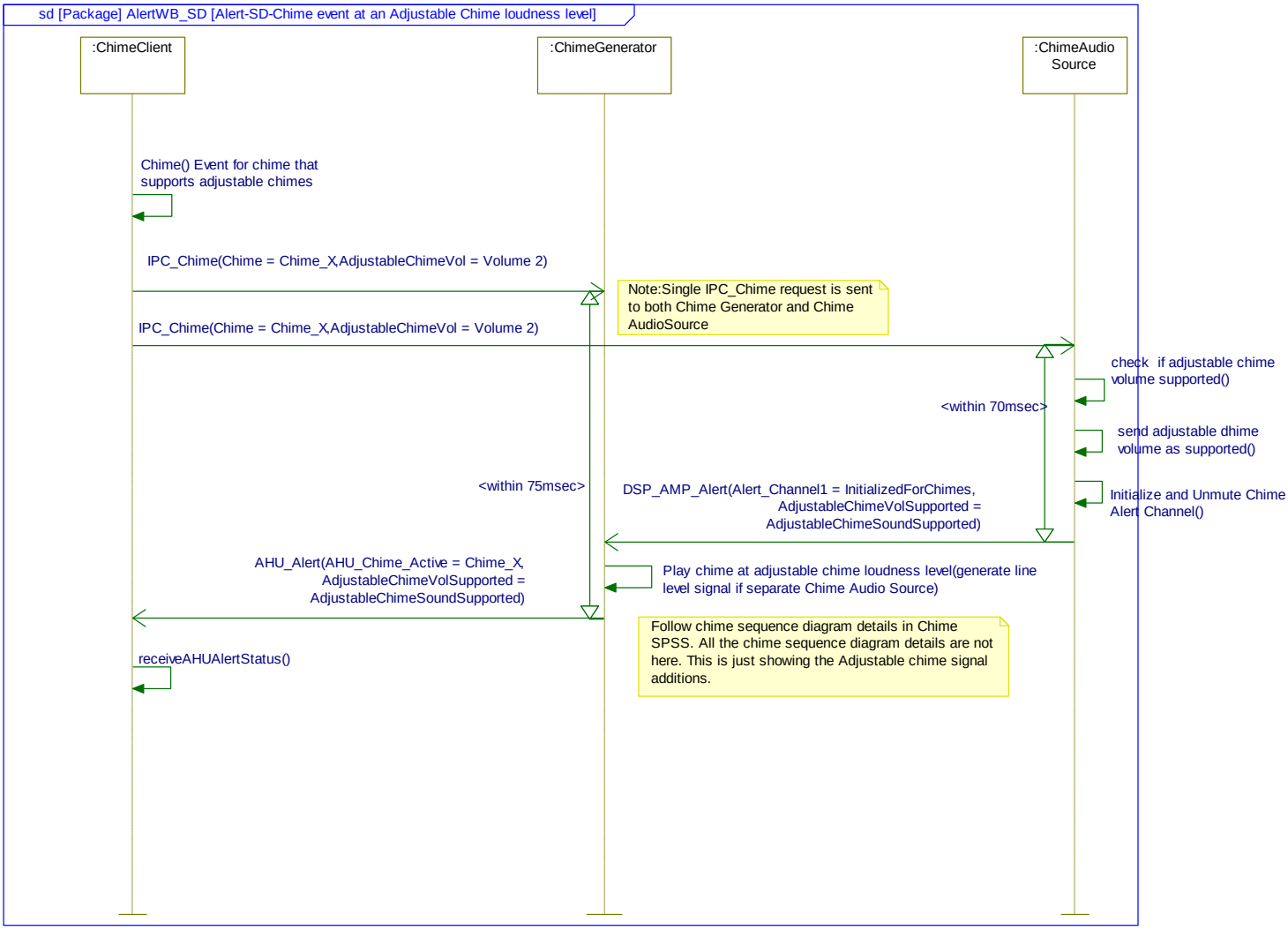
On a previous ignition cycle adjustable Chime_X was set to +2 adjustable chime volume

Event:

A chime event occurs on the vehicle in which Chime_X is sent from the Cluster to the infotainment system with the Chime_X adjustable chime volume set to +2

Post-Condition:

The chime is played by the infotainment system at the +2 level indicated in the Adjustable Chime signal from the Cluster



3.3 ALERT-FUN-REQ-348785/A-Mixable Prompts - Variant 2

3.3.1 Requirements

3.3.1.1 ALERTv2-SR-REQ-348824/A-Prompt Generator and Prompt Audio Source

The Prompt Generator functionality:

1. Generates the prompts. If DSP AMP variant2 is present, then the prompt generator sends the prompt audio signal to the DSP AMP variant 2
2. Is responsible for the audio attenuation of the non-prompt audio sources while a prompt is active
3. Is responsible for volume control
4. Tells the Prompt Audio Source to initialize for prompts within $T_{prompt_initializaiton}$ using the signal 'Alert_Chan = Initialize for Prompts'.
5. Tells the Prompt Audio Source what speakers to play the prompts through using the Prompt_Directionality signal.
6. The Prompt Generator shall keep Prompt_Directionality = Directionality_X for the duration of the prompt.
7. The Prompt Generator shall keep Alert_Chan = Initialize for Prompts for the duration of the prompts.
8. After sending the Alert_Chan = Initialize_For_Prompts the Prompt Generator shall generate the prompt after receiving DSP_Alert_Channel = Initialized_For_Prompts from the Prompt Audio Source.
    - If the Prompt Generator does not receive DSP_Alert_Channel = Initialized_For_Prompts within $T_{prompt_initializaiton}$ + 30 msec of sending Alert_Chan = Initialize_For_Prompts, then the Prompt Generator shall generate the prompt anyways after $T_{prompt_initializaiton}$ + 30 msec.

The Prompt Audio Source functionality:

1. Is responsible for playing the prompts through the speakers indicated in the Prompt_Directionality signal
2. Is responsible for mixing the prompt audio with the active audio source

3.3.1.2 ALERT-TMR-REQ-014782/D-Tprompt_initialization (TcSE ROIN-41488-2)

Name	Description	Units	Range	Resolution	Default
Tprompt_initialization	Maximum time allowed from when the Prompt Generator requests the Prompt Audio Source to initialize for prompts until the Prompt Audio Source responds to the Prompt Generator that it is initialized for prompts and capable of receiving the Prompt Generated signal. Note: use the default value	msec	0-1000	10	50

3.3.1.3 ALERT-SR-REQ-014783/F-Prompt Audio Source response to prompt request (TcSE ROIN-41512-5)

The volume settings sources are Media, Phone, Prompts, VR and TA. Once the Prompt Audio Source is initialized for prompts (Alert_Chan = Initialized for prompts) then prompts are the current active volume setting source. This means if there are user volume adjustments that prompts will be updated by the adjustment since it is the active volume settings source and the other volume settings sources will not be updated unless noted otherwise.

When prompts are the active volume setting source with the other audio sources attenuated (audio sources that can be attenuated during prompts are defined in requirement "VOL-FUR-REQ-014819-Volume Attenuation/Restoration (TcSE ROIN-27919-9)") then since volume user adjustments only adjust the non-attenuated prompt volume the partial attenuation of the attenuated sources would not be cancelled by volume adjustments (ex. Media attenuation wouldn't be cancelled during prompts).

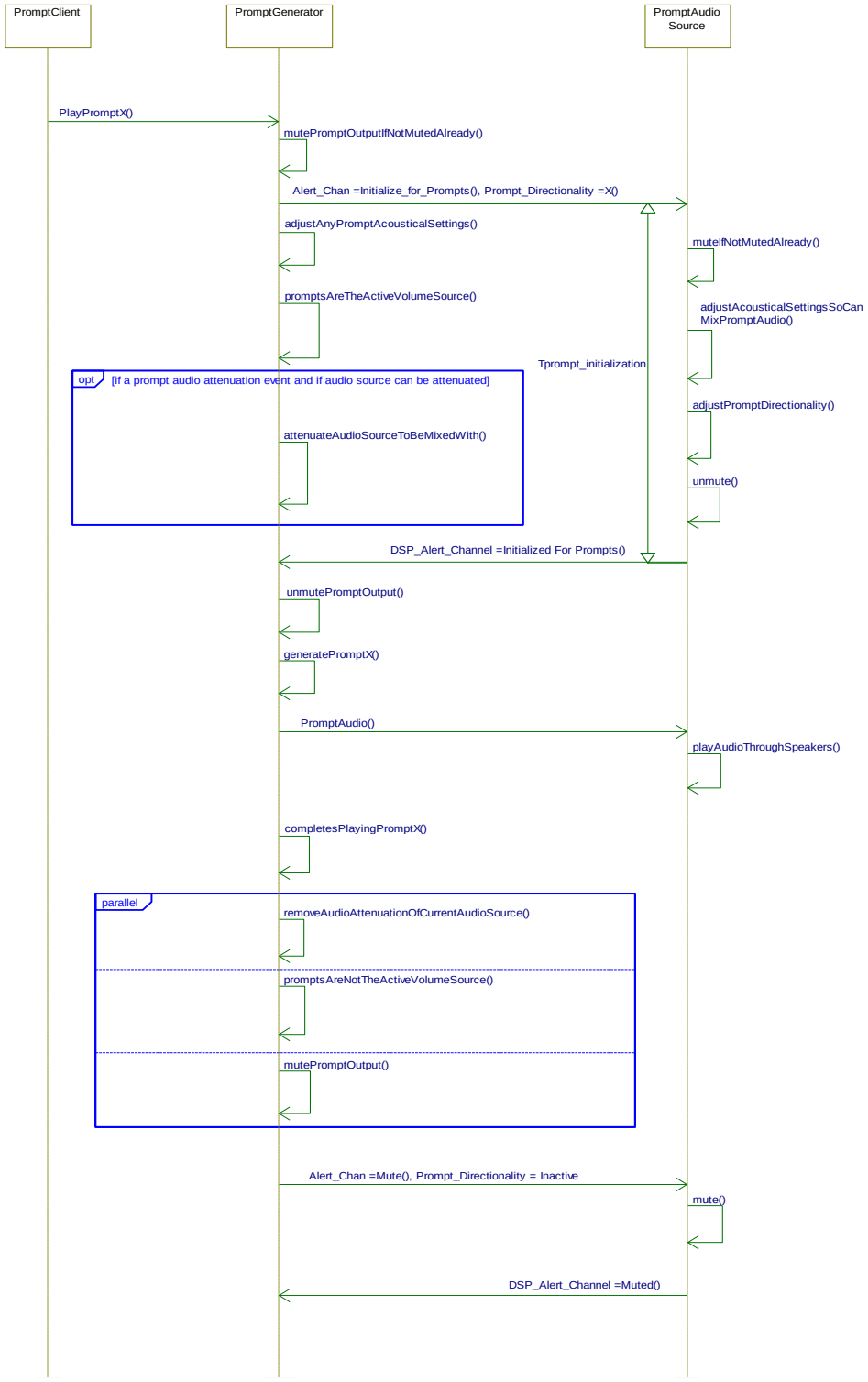
Once the Prompt Audio Source is no longer initialized for prompts then the active volume settings source will not be prompts but the active volume source indicated in the ResourceUpdate.St message. See volume section for further details. Note: this means when SYNC_Alerts : Alert_Chan = "Inactive" or "Mute" the ResourceUpdate signal is used for the volume settings source.

Note: when using this method for mixable prompts (ie SYNC_Alerts in this function) then this method for the Prompt Generator would be used for mixable prompts in place of the prompt strategy in the audio management feature. Unless noted otherwise this method should be used for the prompt generator.

3.3.2 Sequence Diagrams

3.3.2.1 ALERTv2-SD-REQ-348799/A-Prompt Activation

Pre-Condition:
No prompts are active


PromptClient
PromptGenerator
PromptAudio Source
PlayPromptX()
mutePromptOutputIfNotMutedAlready()
Alert_Chan =Initialize_for_Prompts(), Prompt_Directionality =X()
adjustAnyPromptAcousticalSettings()
muteIfNotMutedAlready()
promptsAreTheActiveVolumeSource()
adjustAcousticalSettingsSoCan MixPromptAudio()
Tprompt_initialization
opt
[if a prompt audio attenuation event and if audio source can be attenuated]
adjustPromptDirectionality()
attenuateAudioSourceToBeMixedWith()
unmute()
DSP_Alert_Channel =Initialized For Prompts()
unmutePromptOutput()
generatePromptX()
PromptAudio()
playAudioThroughSpeakers()
completesPlayingPromptX()
parallel
removeAudioAttenuationOfCurrentAudioSource()
promptsAreNotTheActiveVolumeSource()
mutePromptOutput()
Alert_Chan =Mute(), Prompt_Directionality = Inactive
mute()
DSP_Alert_Channel =Muted()

3.4 ALERTv2-FUN-REQ-095181/B-Beep

For enabling / disabling touch panel beeps and for touch panel beeps configuration reference vehicle settings functions "VS-FUN-REQ-025233-Touch Panel Beeps Settings (TcSE ROIN-292335-1)".

3.4.1 Beep Activation Requirements

For certain events (ex. touch sense press event) the infotainment system may need to produce audible beeps.

3.4.1.1 ALERT-SR-REQ-014786/C-Beep Generator allowable times for producing Beeps (TcSE ROIN-273468-1)

The Beep Generator shall be able to produce beeps for the EFP (FCIM) and APIM (FCDIM) when HMIAudioMode = ON and shall not produce beeps when HMIAudioMode = OFF.

~~The Beep Generator shall be able to produce beeps for the OHCM when Demand_PwrModing = ON~~

3.4.1.2 ALERT-REQ-014787/B-EFP Beep Requests (TcSE ROIN-304907-1)

For EFP(FCIM) that support beeps the EFP(FCIM) shall NOT send the network signal to produce an audio beep for selection of climate control inputs unless climate functions are active in the current Vehicle State (Climate Indicators State = Enabled as defined within the Powering Moding section of the applicable Climate Control Functional Specification) i.e. climate functions may not be active when Ignition does not equal Run.
NOTE: The EFP(FCIM) shall continue to send the network signal to produce audio beeps for all infotainment inputs (ex seek touch sense button) regardless of climate control input status.

The EFP(FCIM) shall send infotainment beep requests whenever HMIAudioMode = ON / Multimedia_System = ON.

3.4.1.3 ALERT-SR-REQ-014788/B-Beep Directionality (TcSE ROIN-273469-1)

The Beep Generators / Beep Audio Source shall produce the infotainment beeps out of the front speakers

3.4.1.4 ALERTv2-SR-REQ-095182/D-Infotainment System Beep Set-up

The Beep Generator and Beep Audio Source will vary depending on the infotainment system module availability. The Infotainment System Beep set-up will follow the table below.

Modules Present	Beep Client	Beep Generator	Beep Audio Source	Comments
AHU / DSP AMP / APIM / touch sense EFP	APIM / EFP	DSP AMP	DSP AMP	DSP AMP produce audible beep based on infotainment network message from Beep Client
AHU / APIM / touch sense EFP	APIM / EFP	AHU	AHU	AHU produce audible beep based on infotainment network message from Beep Client

Beep Configuration Table 1

Modules Present	Beep Client	Beep Generator	Beep Audio Source	Comments
IAHU / DSP AMP	IAHU	DSP AMP	DSP AMP	DSP AMP produce audible beep based on infotainment

The information contained in this document is Proprietary to Ford Motor Company.

				network message from Beep Client
IAHU	IAHU	IAHU	IAHU	

Beep Configuration Table 2

Beep Configuration Table 2

Modules Present	Beep Client	Beep Generator	Beep Audio Source	Comments
IAHU / / DSP AMP v2	IAHU	IAHU	IAHU	IAHU produces the beep and sends it over A2B as a chime for the front speakers
IAHU	IAHU	IAHU	IAHU	

IAHU – integrated system master and AHU
EFP – CAN based button panel (touch sense)
Note: DSP AMP and DSP AMP variant 2 are mutually exclusive

3.4.1.5 ALERT-TMR-REQ-014790/D-T_Beep_Event (TcSE ROIN-110901-1)

Name	Description	Units	Range	Resolution	Default
T_Beep_Event	The maximum time allowed from when a beep event begins until the signal 'XXX_Audible_Beep = Active' is put on the infotainment bus by the beep client. Note: use default value	msec	0-1000	10	30

3.4.1.6 ALERT-SR-REQ-014791/E-Beep Activation (TcSE ROIN-39866-4)

When there is both an AHU and DSP AMP on the vehicle at the same time then the DSP AMP shall become the Beep Generator. The Beep Client(s) shall send the Beep Generator the infotainment network signal 'Audible_Beep = Active' to produce an audio beep (ex. touch sense button press beeps). The Beep Generator shall produce the audible beep within T_Audible_Beep of receiving the signal 'Audible_Beep = Active'.

After the Beep Generator produces the beep it shall wait for the signal 'Audible_Beep to equal 'Inactive' before it produces another beep based on the signal 'Audible_Beep ' set to 'Active'.

Note: The DSP AMP and DSP AMP variant 2 are mutually exclusive. The DSP AMP variant 2 is not the Beep Generator (the IAHU is).

3.4.1.7 ALERT-TMR-REQ-014792/E-T_Audible_Beep (TcSE ROIN-56911-3)

Name	Description	Units	Range	Resolution	Default
T_Audible_Beep	The maximum time allowed from when the Beep Generator receives the 'Audible_Beep == Active' signal until an audible beep is produced. Note: use default value	msec	0-1000	10	50

The information contained in this document is Proprietary to Ford Motor Company.

3.4.2 Sequence Diagrams

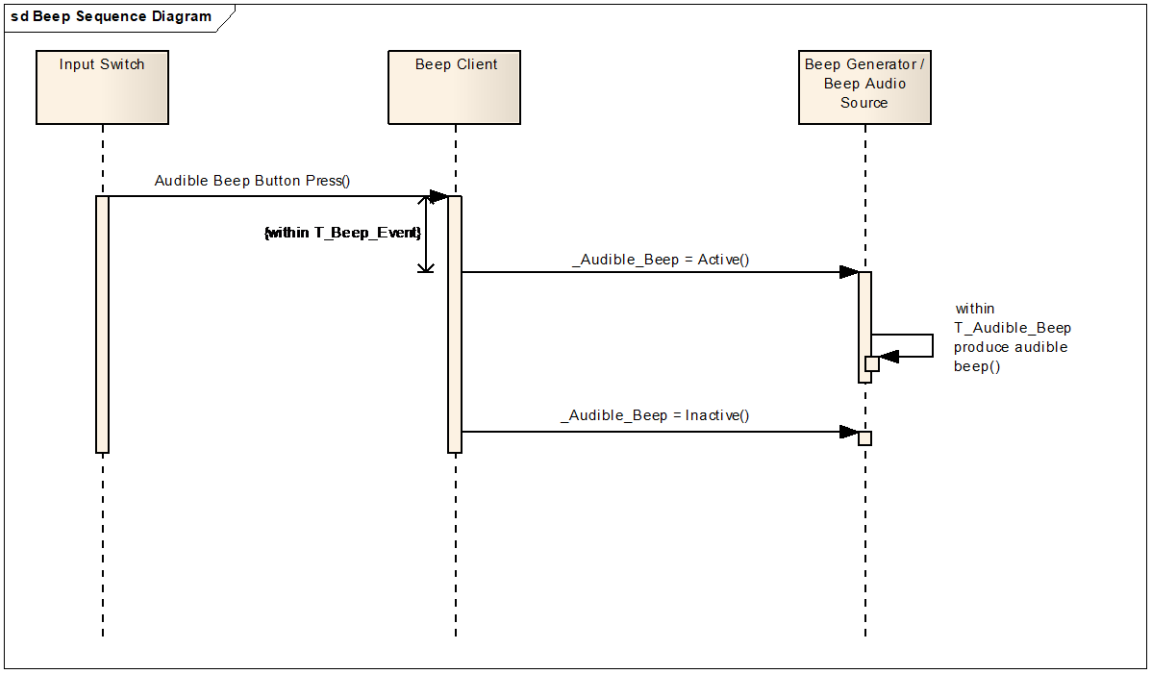
3.4.2.1 ALERT-SD-REQ-014793/B-Beep Activation (TcSE ROIN-262985-1)

Pre-condition
No beep active and beeps are supported

Scenario
Audible Beep event (ex touch sense button press event)

Post-condition
Audible Beep is produced

Sequence Diagram



3.5 ALERT-FUN-REQ-014794/C-Audio Attenuation/Muting Strategy (TcSE ROIN-119789-1)

There may be multiple audio attenuation / muting requests from outside of the Infotainment System (ex. FCW, RPA…) but the AHU / DSP AMP shall only support 1 audio attenuation / muting request signal from outside the Infotainment System. The 'IPC_Infotainment.St() : Attn_Info_Audio' signal shall be used to Attenuate / Mute the infotainment audio. The Cluster is the Audio Attenuation Request Prioritizer responsible for sending the audio attenuation requests from outside the Infotainment System to the AHU / DSP AMP.

- Note: Regardless of whether the Chime_Source is the Cluster or Infotainment System the Audio Attenuation Requests are independent of chimes and chime source and shall always be supported.

There may also be muting requests internal to the infotainment system such as for mixable prompts using the 'SYNC_Alerts : Attn_Info_Audio' signal.

3.5.1 Cluster Prioritizing Audio Attenuation Requests

3.5.1.1 ALERT-SR-REQ-014795/C-Cluster Prioritizing Audio Attenuation Requests (TcSE ROIN-39883-2)

The Cluster may receive multiple audio attenuation requests at the same time from outside the infotainment system. While multiple audio attenuation requests are active at the same time the Cluster shall use the request with the greatest attenuation of audio for the Attn_Info_Audio signal. The Cluster Attn_Info_Audio signal shall be broadcast on the Infotainment bus.

For example, while FCW (forward collision warning) is active with a full audio mute if a Reverse Park Aid signal is requesting the audio be partially attenuated then the audio will remain fully muted since FCW has the highest attenuation level.

Upon exiting the audio attenuation / muting request with the greatest attenuation, the next highest audio attenuation request shall be serviced until there are no longer any active audio attenuation requests.

3.5.2 AHU / DSP AMP Prioritizing Audio Attenuation Requests

3.5.2.1 ALERT-SR-REQ-014796/C-Multiple Attenuation Requests (TcSE ROIN-39884-2)

The AHU / DSP AMP may receive multiple audio attenuation requests at the same time whether from outside the infotainment system (ex. chime requests sent to the Cluster which sends attenuation requests to infotainment components) or internal to the infotainment system (ex. SYNC prompts attenuation requests). While multiple audio attenuation requests are active at the same time the AHU / DSP AMP shall use the request with the greatest attenuation of audio.

Upon exiting the audio attenuation / muting request with the greatest attenuation, the next highest audio attenuation request shall be serviced until there are no longer any active audio attenuation requests.

3.5.3 Infotainment System Audio Attenuation

3.5.3.1 ALERT-SR-REQ-014797/C-Audio Attenuation to Attenuation_X volume level (TcSE ROIN-39889-2)

When the AHU / DSP AMP receives the signal Attn_Info_Audio with New_Attn_Event = Active then the AHU / DSP AMP will lower their infotainment volume to Attenuation_X audio level as defined in the AHU / DSP AMP component requirements (see requirement VOL-GREQ-27919-6-Volume Attenuation/Restoration).

The Transmitter Audio Attenuation Client shall set Attn_Info_Audio to the desired attenuation level for the duration of the requested audio attenuation. See ALERT-GREQ-39896-3-Transmitters usage of New_Attn_Event signal for the usage of the New_Attn_Event signal.

Ex.
1. Audio Attenuation Client sends an attenuation request with Attn_Info_Audio = Attenuation_X / New_Attn_Event = Active
2. Audio Attenuation Client sends Attn_Info_Audio = Attenuation_X / New_Attn_Event = Inactive (See requirement 39896 Transmitters usage of New_Attn_Event for details).
3. Audio Attenuation Client keeps sending Attn_Info_Audio = Attenuation_X / New_Attn_Event = Inactive while the attenuation event is occuring (note: Audio Attenuation Client has no knowledge if a partial attenuation is cancelled).
4. When Audio Attenuation Client no longer needs the attenuation it shall send Attn_Info_Audio = 'No Attenuation of Audio" / New_Attn_Event = Inactive.

3.5.3.2 ALERT-SR-REQ-014798/C-Audio Attenuation Timing (TcSE ROIN-39890-2)

The AHU / DSP AMP shall lower their volume to the predefined audio level within 100 msec of receiving the Attn_Info_Audio signal with the New_Attn_Event = Active if the Attn_Info_Audio signal causes an attenuation event.

The AHU / DSP AMP shall lower their volume assuming that the present audio volume level is higher than the Attenuation_X level, otherwise the audio volume and XXX_Volume_Level signal shall not change.

3.5.3.3 VOL-SR-REQ-014799/C-Volume signal update for Audio Attenuation (TcSE ROIN-39891-2)

The AHU / DSP AMP shall update the XXX_Volume_Level signal after receiving the Attn_Info_Audio signal that results in a decrease in audio.

Note: the corresponding XXX_Volume_Updated signal with the XXX_Volume_Level signal shall be set to "No Update" for an Attn_Info_Audio attenuation event that changes the XXX_Volume_Level signal.

3.5.3.4 ALERT-SR-REQ-014800/C-User Volume adjustment lock-out period (TcSE ROIN-39892-3)

During an Attn_Info_Audio audio attenuation event volume user adjustments won't be able to lower or increase volume ~~shall be ignored~~ until 200 msec after the AHU / DSP AMP first receives the Attn_Info_Audio signal set to an attenuation level with the signal New_Attn_Event = Active.

Only after 200 msec from first receiving Attn_Info_Audio = Attenuation_X / New_Attn_Event = Active shall volume user adjustments cancel the partial attenuation.

3.5.3.5 ALERT-SR-REQ-014801/E-Volume user adjustments during a partial attenuation (TcSE ROIN-39893-4)

Volume user adjustments are allowed after 200 msec from first receiving the Attn_Info_Audio signal with the signal New_Attn_Event = Active. Volume user adjustments to any of the attenuated volume sources shall effectively cancel the active partial attenuation attenuating the affected volume sources.

Ex.
1. Pre-Condition: XXX_Volume_Level = 20
2. Audio Attenuation Clients sends a partial Attenuation request with Attenuation_X / New_Attn_Event = Active which sets the volume level to volume step 5
3. Then the Audio Attenuation Client sets Attn_Info_Audio = Attenuation_X / New_Attn_Event = Inactive
4. The Audio Attenuation Server volume goes to XXX_Volume_Level = 5 and XXX_Volume_Updated = No Update.
5. The user adjust the attenuated volume up with SetVolume = +1 step while the partial attenuation is active
6. The partial attenuation is then cancelled and volume is increased to volume step 6 (XXX_Volume_Level = 6)

Note: If two partial attenuations are active at once then cancelling the partial attenuation of one partial attenuation doesn't necessarily cancel the attenuation of another partial attenuation.

Ex.
Pre-Condition:
1. There is both a Chime (IPC_Infotainment) AND Prompt (SYNC_Alerts) partial attenuation event active at the same time.
2. A media source (ex AM/FM) is granted in the Resource Update message and Prompts are active. Since prompts are active they are the active volume source

Event:
User adjust the volume while prompts are active

Post-Condition:
The chimes partial attenuation (IPC_Infotainment) is cancelled so prompts are no longer attenuated but the media source still is attenuated from the Prompt partial attenuation (SYNC_Alerts). The prompt partial attenuation is active so media is still attenuated. The prompt partial attenuation is still active per requirements 41512 and 27919.

3.5.3.6 ALERT-SR-REQ-014802/C-Volume user adjustments during a full mute (TcSE ROIN-39894-2)

During an Attn_Info_Audio attenuation event volume user adjustments shall be ignored while a full mute Attenuation event is active unless otherwise noted in the AHU / DSP AMP component requirements. The AHU / DSP AMP shall update the XXX_Volume_Level signal accordingly.

3.5.3.7 ALERT-SR-REQ-014803/C-New Attenuation request at the same attenuation level that was cancelled because of a volume user adjustment (TcSE ROIN-39895-2)

To protect for the case were another Attn_Info_Audio attenuation event happens requesting a partial attenuation at the same Attn_Info_Audio attenuation level that the AHU / DSP AMP is ignoring (because of a previous user volume adjustment) then to make sure the attenuation is still acted on the AHU / DSP AMP shall monitor the "New_Attn_Event".

If the AHU / DSP AMP receive the "New_Attn_Event = Active" concurrently with "Attn_Info_Audio" signal set to an attenuation level then the audio shall be attenuated (if applicable per other attenuation requirements) even if "Attn_Info_Audio" signal attenuation level did not change.

3.5.3.8 ALERT-SR-REQ-014804/C-Transmitters usage of New_Attn_Event signal (TcSE ROIN-39896-3)

The Audio Attenuation Transmitting module shall set the "New_Attn_Event = Active' when it requests a new Attenuation event.

The "New_Attn_Event" signal shall be set equal to "Active" and then set to "Inactive" (within 30 msec) and then all subsequent periodic status updates shall set the signal "New_Attn_Event = Inactive" until a new Attenuation event occurs.

Ex. Audio Attenuation Client requesting the same audio attenuation at different times:
1. The Audio Attenuation Client (audio attenuation transmitting module) requests a new attenuation event when a Reverse Park Aid chime begins. The Audio Attenuation Client sends a partial attenuation request with Attn_Info_Audio = Attenuation_X and New_Attn_Event = Active.

2. Then the Audio Attenuation Client keeps Attn_Info_Audio equal to Attenuation_X and sets New_Attn_Event = Inactive

3. the Audio Attenuation Client keeps Attn_Info_Audio equal to Attenuation_X for as long as the Reverse Park Aid beeps are playing but keeps New_Attn_Event = Inactive

4. The Audio Attenuation Client requests the Seat Belt chime to play out of the front speakers while the Reverse Park Aid chime continues to play out of the rear speakers. The Audio Attenuation Clients sends a partial attenuation request with Attn_Info_Audio = Attenuation_X and New_Attn_Event = Active.

5. Then the Audio Attenuation Client keeps Attn_Info_Audio equal to Attenuation_X and sets New_Attn_Event = Inactive

6. the Audio Attenuation Client keeps Attn_Info_Audio equal to Attenuation_X for as long as the Seat Belt chime or Reverse Park Aid chime are playing but keeps New_Attn_Event = Inactive

7. After some time the Reverse Park Aid chime ends and the Audio Attenuation Client keeps Attn_Info_Audio = Attenuation_X and keeps New_Attn_Event = Inactive.

8. Some time after the Reverse Park Aid chime ends the Seat Belt chime ends and the Audio Attenuation Client sets Attn_Info_Audio = 'No Attenuation of Audio' and keeps New_Attn_Event = Inactive.

Note: The seat belt chime and RPA chime are used as examples and may or may not be the same attenuation level. Please reference the applicable function specifications for what attenuation to use

3.5.3.9 ALERT-SR-REQ-014805/G-Volume when changing to a lower attenuation / no attenuation from a higher attenuation (TcSE ROIN-39897-2)

When the 'Attn_Info_Audio = Attenuation_X' signals change to a lower attenuation level or to no attenuation resulting in an increase in volume then the AHU / DSP AMP shall gracefully increase the volume at a rate defined in Volume SPSS requirement “FUR-REQ-088208-Audio Attenuation / Mute Ramps”.

Upon exiting the audio attenuation / muting request with the greatest attenuation, the next highest audio attenuation request shall be serviced until there are no longer any active audio attenuation requests.

Example from two different modules sending different attenuation values in different network messages:
1. Pre-Condition: XXX_Volume_Level = 20

2. Audio Attenuation Client A sends a partial Attenuation request with Attenuation_X/ New_Attn_Event = Active which sets the volume level to volume step 16 (see attenuation table for actual volume steps to use – ie VOL-REQ-014819-Volume Attenuation/Restoration)
3. Then the Audio Attenuation Client A sets Attn_Info_Audio (Module A) = Attenuation_X/ New_Attn_Event = Inactive
4. The Audio Attenuation Server volume goes to XXX_Volume_Level = 16 and XXX_Volume_Updated = No Update.
5. Audio Attenuation Client B sends a partial Attenuation request with Attenuation_Y / New_Attn_Event = Active which sets the volume level to volume step 7
6. Then the Audio Attenuation Client B sets Attn_Info_Audio (Module B) = Attenuation_Y / New_Attn_Event = Inactive
7. The Audio Attenuation Server volume goes to XXX_Volume_Level = 7 and XXX_Volume_Updated = No Update.
8. Then the Audio Attenuation Client B sets Attn_Info_Audio (Module B) = No Attenuation of Audio / New_Attn_Event = Inactive
9. The Audio Attenuation Server volume goes to XXX_Volume_Level = 16 and XXX_Volume_Updated = No Update.
10. Then the Audio Attenuation Client A sets Attn_Info_Audio (Module A) = No Attenuation of Audio / New_Attn_Event = Inactive
11. The Audio Attenuation Server volume goes to XXX_Volume_Level = 20 and XXX_Volume_Updated = No Update.

3.5.3.10 ALERT-SR-REQ-014806/C-Timing to increase volume because of a lowering of the attenuation level (TcSE ROIN-39898-2)

When the 'Attn_Info_Audio = Attenuation_X' signals change to a lower attenuation level or to no attenuation resulting in an increase in volume then the ACM / DSP AMP shall start to increase the volume level no later then 200 msec after receiving the Attenuation update.

4 Appendix: Reference Documents

Reference #	Document Title
1	Chime sound characteristics used to be in AHU Hardware spec pre SYNC 4.1. Get equivalent spec from core audio for SYNC 4.2
2	Reference SYNC 4.2 IDS for chime diagnostics
3	Reference SYNC internal specs regarding what mixable prompts to play and when. Reference A22b and A22c
4	When Cluster is integrated with SYNC then reference the Cluster chime arbitrator specification for additional Cluster functionality
5	HMI specs if adjustable chimes are supported for adjustable chime HMI
6	
7	
8	
9	
10	
11	
12	
13	
14	