



Research & Vehicle Technology
“Infotainment Systems Product Development”

Feature – V2I LITE Application

**Infotainment Subsystem Part Specific
Specification (SPSS)**

Version 1.4.17

UNCONTROLLED COPY IF PRINTED

Version Date: Jan 2023

FORD CONFIDENTIAL



Revision History

Date	Ver	Notes	Approver by
May. 17, 2021	1.0	Initial Release	Li,yifei
	Mengmeng Lin (mlin37)	4.1 V2I Setting SDK	
	Yuanyan Zhang(yzhan295)	4.2 Audio Channel Arbitration	
	Hang Gong(hgong17)	4.3 Text-to-Speech 4.4 CAN Signal Receiver	
	Chengxu Liu(cliu183)	4.5 CAN Signal Transmitter	
	Jinxian Wang(Jwang386)	4.6 SYNCP	
	Chengxu Liu(cliu183)	4.7 Embedded Navigation	
	Yuanyan Zhang(yzhan295)	4.8 System Configuration, IPPT, WIR	
May. 17, 2021	1.1	Update Below	
	Chengxu Liu(cliu183)	添加了 CTL 部分的接口定义和说明	
June. 7, 2021	1.2	Update Below	
	Chengxu Liu(cliu183)	更新了 4.7.4.16 接口	
	Jinxian Wang(Jwang386)	更新了 4.2 接口	
June. 23, 2021	1.3	Update Below	
	Gong Hang	把 getTheme 接口单独从 getSystemInfo 分离出来并自成一章	
June. 25, 2021	1.3.1	Update Below	
	Gong Hang	把 System Configuration , WIR 拆分成 4.8 V2I-LITE-FUN-REQ-480001/A- self-initialization/ 4.9 V2I-LITE-FUN-REQ-490001/WIR Policy Table/ 4.10 V2I-LITE-FUN-REQ-4100001/A-IP Whitelist 三个章节	
June. 25, 2021	1.4	Update Below	
	Gong Hang	在 4.8 开机自启动章节中加入了 YFV 的建议方案	
	Chengxu Liu(cliu183)	在 4.14 获取主题章节增加了 callback 接口	
June. 30, 2021	1.4.1	Update Below	Update Below
	Gong Hang	<ul style="list-style-type: none"> 在 4.5 章中的 usecase 中增加了”保证数据送达, 并要求不低于 100m 延迟。详细指标参考 CAN 网络要求” 在 4.6 章中 usecase 中增加了”保证 1s 内可以成功完成 24 次数据的加解密。” 	
June. 30, 2021	1.4.2	Update Below	
	Gong Hang	更新了 4.1 章节的示例代码	
July. 6, 2021	1.4.3	Update Below	
	Gong Hang	增加了 4.17 章节 APK installation & Configuration	
Sep.8,2021	1.4.4	Update Below	
	Gong Hang	在 4.8 章节, 修改需求 “需要系统提供一种方式保证 V2I APP 能够在开机后两分钟内完成启动”	
Nov.24,2021	1.4.5	Update Below	
	Zhang Wei (wzhan175)	在 1.4.4 章节, 增加备注 “注: Phase5 与 Phase4 保持一致, 不做修改; 待数据质量提高后, TBD。”	
		在 4.7.4.8 章节, 修改需求: 参数类型 Imageview 改成 Bitmap	
		在 4.9.1 章节, 增加需求: 新增 “Package Name for WIR policy”	
		在 4.10.1 章节, 增加需求: 新增服务城市 “北京” 和 “广州”	
Jan.18,2022	1.4.6	Update Below	
	Zhang Wei(wzhan175)	补充 phase4 已有需求: 4.3.2.6 V2I-LITE-REQ-430326/ void ITsListener.onSpeechStop(int status, String message)	
Mar.2,2022	1.4.7	Update Below	
	Zhang Wei(wzhan175)	在 4.1.2 章节, 增加需求:	



		4.1.2.13 V2I-LITE-REQ-4101213/A-Apply V2I Favorite Operation 4.1.2.14 V2I-LITE-REQ-4101214/A-Cancel V2I Favorite Operation 4.1.2.15 V2I-LITE-REQ-4101215/A-Query V2I Favorite state 删除需求: 4.1.2.11 V2I-LITE-REQ-4101211/A-Request to show the V2I in favorite	
		在 1.1.4.5 章节, 添加接口调用说明 在 4.1.4 章节, 增加接口: 4.1.4.13 boolean applyV2IFavorite() 4.1.4.14 boolean cancelV2IFavorite() 4.1.4.15 int queryV2IFavoriteState() 删除接口: 4.1.4.11 boolean canShowInFavorite()	
Mar.7,2022	1.4.8	Update Below	
	Zhang Wei(wzhan175)	增加需求: 4.18 V2I-LITE-FUN-REQ-4180001/A-TEE	
Mar.9,2022	1.4.9	Update Below	
	Zhang Wei(wzhan175)	更新需求: 4.10 V2I-LITE-FUN-REQ-4100001/A-IP Whitelist	
Mar.15,2022	1.4.10	Update Below	
	Zhang Wei(wzhan175)	更新需求: 4.11.3.2 Sequence Diagrams	
June.17,2022	1.4.11	Update Below	
	Zhang Wei(wzhan175)	更新需求: 4.1.4.7 void requestV2ITrial()增加参数: requestV2ITrial(boolean extendFlag, string date)	
June.27,2022	1.4.13	Update Below	
	Zhang Yuanyan/Lin Mengmeng/Zhang, Wei	在 4.1.4.7 章节更新需求: 4.1.4.7 void requestV2ITrial(string date) 在 4.1.4.16 章节, 增加需求: 4.1.4.16 string queryV2IAuthDate() 在 4.1.4.17 章节, 增加需求: 4.1.4.17 void updateV2IAuthDate (string date)	
Aug.16,2022	1.4.14	Update Below	
	Zhang, Wei	在 4.1.4.18 章节, 增加需求: 4.1.2.18 V2I-LITE-REQ-4101217/A-Pop up Toast 4.1.2.19 V2I-LITE-REQ-4101217/A-Page Change 4.1.4.18 void onToast (String message) 4.1.4.19 void onPageChange(SettingPage settingPage)	
Sep. 16, 2022	1.4.15	Update Below	
	Zhang, Wei	在 4.4.4.5 章节, 更新 CAN Receiver Signals' List	
Dec. 14, 2022	1.4.16	Update Below	
	Zhang, Wei	1. 在 4.1.4.18 章节, 删除 queryV2ITrialState 调用失败 toast 2. 删除 4.1.4.16 V2I-LITE-FUN-REQ-4160001/A-verifyCtl (Phase5 采用 Certificate Pinning 方案, 从 Pin leaf certificate (有效期 3 年) 修改为 root certificate (有效期 70 年)) 3. 更新 4.5 V2I-LITE-FUN-REQ-450001/A-CAN Signal Transmitter 关于 Dbus 系统架构图和数据格式。	
Jan. 6, 2023	1.4.17	Update Below	
	Zhang, Wei	4.4.4.5 String getXXX() 1) GearLvr_D_ActlDrv 有误, 改为 GearLvrPos_D_Actl; 2) LaActvStats_D_Dsply 改为 LaActvStats_D2_Dsply 4.5 V2I-LITE-FUN-REQ-450001/A-CAN/Dbus Signal Transmitter 1) 更新信号发送相关需求	



DRAFT



Table of Contents

REVISION HISTORY.....	2
1 OVERVIEW.....	8
1.1 Terms and Definitions.....	8
1.2 Assumptions.....	8
2 ARCHITECTURAL DESIGN.....	9
2.1 System Boundary Diagram.....	9
2.1.1 V2I_LITE-IBD-REQ-XXXX/B-V2I_LITE System Service Boundary Diagram.....	9
2.2 System Service Diagram.....	9
2.2.1 V2I_LITE-IBD-REQ-XXXX/B-V2I_LITE System Service Functions Diagram.....	9
3 GENERAL REQUIREMENTS.....	10
3.1 V2I LITE Application Operation Mode and States.....	10
3.1.1 Nominal Operating Modes State Machine.....	10
3.1.2 V2I_LITE -REQ-395076/B-State Transition Prioritization.....	10
3.1.3 V2I_LITE -REQ-405004/B-Transit to Failure State if the V2I_LITE System Service Stays in Standby State too Long.....	10
3.1.4 V2I_LITE-ACT-REQ-405005/A-Time to Stay in Standby Mode.....	10
3.1.5 V2I LITE Top Level Feature State Machine.....	10
3.2 Non-Functional Requirements.....	10
3.3 Security Requirements.....	10
4 FUNCTIONAL DEFINITION.....	11
4.1 V2I-LITE-FUN-REQ-410001/A-V2I Vehicle Setting.....	11
4.1.1 Use Cases.....	11
4.1.2 Requirements.....	11
4.1.3 White Box View.....	12
4.1.4 Interfaces & Parameters.....	14
4.2 V2I-LITE-FUN-REQ-420001/A-Audio Channel Arbitration.....	19
4.2.1 Use Cases.....	19
4.2.2 Requirements.....	20
4.2.3 White Box View.....	20
4.2.4 Interfaces & Parameters.....	20
4.3 V2I-LITE-FUN-REQ-430001/A-Text-to-Speech.....	23
4.3.1 Use cases.....	23
4.3.2 Requirements.....	23
4.3.3 White Box View.....	23
4.3.4 Interfaces & Parameters.....	24
4.4 V2I-LITE-FUN-REQ-440001/A-CAN Signal Receiver.....	25
4.4.1 Use cases.....	25
4.4.2 Requirements.....	25
4.4.3 White Box View.....	25
4.4.4 Interfaces & Parameters.....	25
4.5 V2I-LITE-FUN-REQ-450001/A-CAN/Dbus Signal Transmitter.....	27
4.5.1 Use cases.....	28
4.5.2 Requirements.....	28
4.5.3 White Box View.....	28
4.5.4 Interfaces & Parameters.....	28



4.6	V2I-LITE-FUN-REQ-460001/A-Keystore Encryption & Decryption	31
4.6.1	Use cases.....	31
4.6.2	Requirements	32
4.6.3	White Box View.....	32
4.6.4	Interfaces & Parameters.....	32
4.7	V2I-LITE-FUN-REQ-470001/A-Embedded Navigation.....	34
4.7.1	Use cases.....	34
4.7.2	Requirements	34
4.7.3	White Box View.....	34
4.7.4	Interfaces & Parameters.....	34
4.8	V2I-LITE-FUN-REQ-480001/A- self-initialization	41
4.8.1	Use cases.....	41
4.8.2	Requirements	41
4.8.3	White Box View.....	41
4.8.4	Interfaces & Parameters.....	41
4.9	V2I-LITE-FUN-REQ-490001/A-WIR Policy Table	41
4.9.1	Use cases.....	41
4.9.2	Requirements	41
4.9.3	White Box View.....	41
4.9.4	Interfaces & Parameters.....	41
4.10	V2I-LITE-FUN-REQ-4100001/A-IP Whitelist.....	42
4.10.1	Use cases.....	42
4.10.2	Requirements.....	42
4.10.3	White Box View.....	42
4.10.4	Interfaces & Parameters.....	42
4.11	V2I-LITE-FUN-REQ-4110001/A-IPPT	43
4.11.1	Use cases.....	43
4.11.2	Requirements.....	43
4.11.3	White Box View.....	44
4.11.4	Interfaces & Parameters.....	45
4.12	V2I-LITE-FUN-REQ-4120001/A-MasterReset	47
4.12.1	Use cases.....	47
4.12.2	Requirements.....	47
4.12.3	White Box View.....	47
4.12.4	Interfaces & Parameters.....	47
4.13	V2I-LITE-FUN-REQ-4130001/A-SYNCP	48
4.13.1	Use cases.....	48
4.13.2	Requirements.....	48
4.13.3	White Box View.....	49
4.13.4	Interfaces & Parameters.....	49
4.14	V2I-LITE-FUN-REQ-4140001/A-getTheme	50
4.14.1	Use cases.....	50
4.14.2	Requirements.....	50
4.14.3	White Box View.....	50
4.14.4	Interfaces & Parameters.....	50
4.15	V2I-LITE-FUN-REQ-4150001/A-getProvisioningState	51
4.15.1	Use cases.....	51
4.15.2	Requirements.....	51
4.15.3	White Box View.....	51
4.15.4	Interfaces & Parameters.....	51
4.16	V2I-LITE-FUN-REQ-4170001/A-APK installation & Configuration.....	52
4.16.1	Use cases.....	52
4.16.2	Requirements.....	52



4.16.3	White Box View.....	52
4.17	V2I-LITE-FUN-REQ-4180001/A-TEE.....	52
4.17.1	Use cases.....	53
4.17.2	Requirements.....	53
4.17.3	White Box View.....	53
4.17.4	Interfaces & Parameters.....	53
5	APPENDIX: REFERENCE DOCUMENTS.....	54

DRAFT



1 Overview

The purpose of the document is to describe and specify each component of V2I/ Vehicle to" feature **V2I LITE – CHN(F006091)**, which will be deployed on SYNC+ in the phase 5 project.

1.1 Terms and Definitions

Abbreviation	Definitions	Description
Setting		Provide a visual page for users to apply for use and set their preferences in IVI
TEE	Trusted execution environment	Separately allocate an isolated memory for sensitive data in the hardware, all sensitive data calculations are performed in this memory, and except for authorized interfaces, other parts of the hardware cannot access the information in this isolated memory. In this way, the privacy calculation of sensitive data can be realized.
WIR	Wireless Interface Router	A flexible enabler service that will support a variety of use cases for connected services, is intended to drive down bulk data costs and enable multi-module cloud networking. Also it provides flexibility to change connectivity policies via Cloud based WIR policy management.
IPPT	IP based Pass Through	Enable a generic approach to preauthorize connection between applications on any Ethernet connected ECUs and 3rd party/Ford applications to retrieve feature specific content for a pre-defined period of time via any connected network interface on the vehicle
SYNCP	SYNC Provision	Provide cryptographic operation to decode or encode a packet.
Provisioning		Provisioning is the process of setting up IT infrastructure. It can also refer to the steps required to manage access to data and resources, and make them available to users and systems. Provisioning shall be responsible for collecting provisioning data from vehicle modules that needed to be provisioned and send to the Provisioning OffBoard Client(such as APIM) and communicate provisioning states back to module

1.2 Assumptions



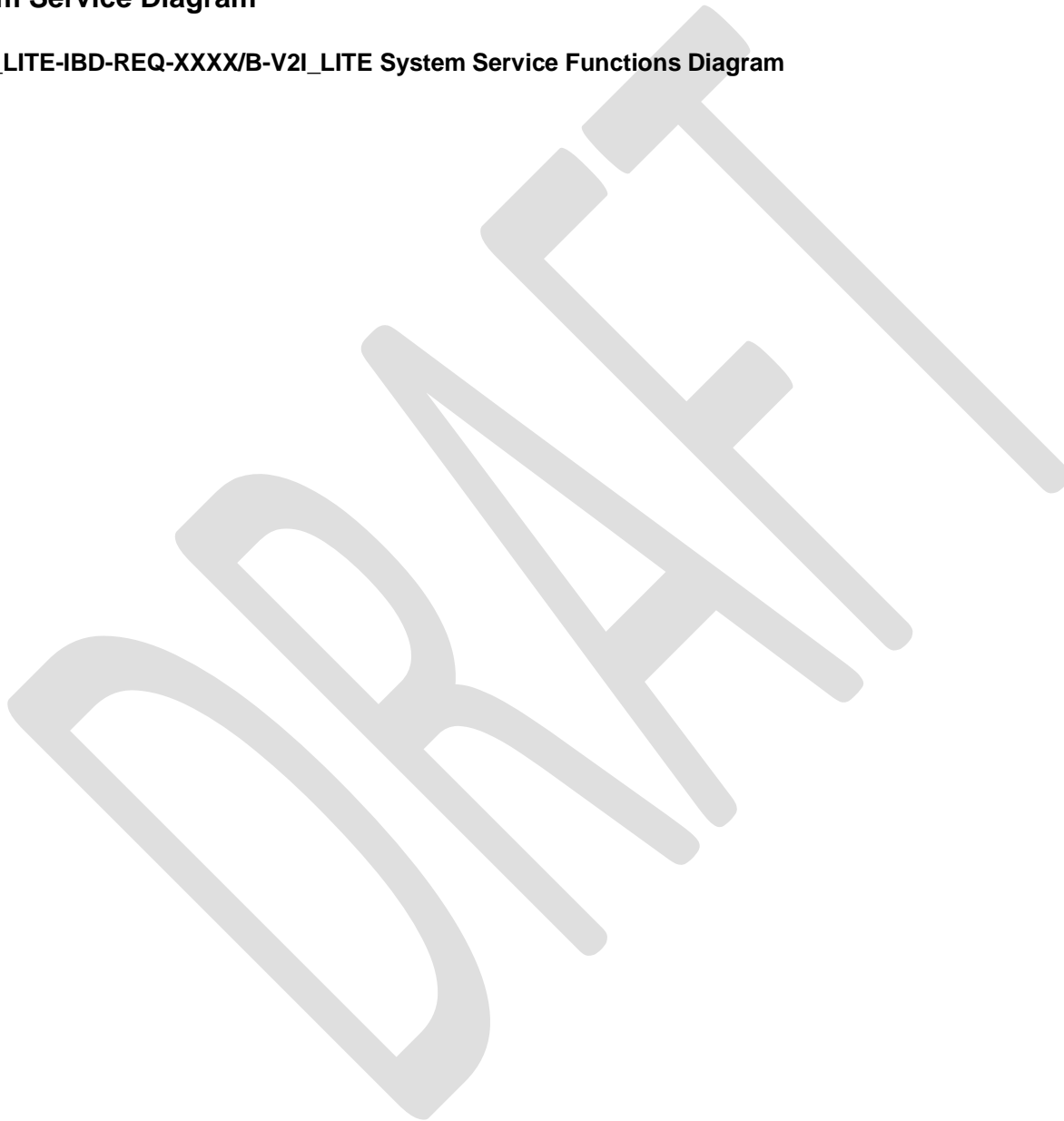
2 Architectural Design

2.1 System Boundary Diagram

2.1.1 V2I_LITE-IBD-REQ-XXXX/B-V2I_LITE System Service Boundary Diagram

2.2 System Service Diagram

2.2.1 V2I_LITE-IBD-REQ-XXXX/B-V2I_LITE System Service Functions Diagram





3 General Requirements

3.1 V2I LITE Application Operation Mode and States

3.1.1 Nominal Operating Modes State Machine

3.1.1.1 *V2I_LITE-STM-REQ-405001/A-Nominal Operating Modes*

3.1.2 V2I_LITE -REQ-395076/B-State Transition Prioritization

The detail of states on Logical Operating Modes can be found below:

State	Description
Startup	
Standby State	
Operational State	
Failure	
Shutdown	

3.1.3 V2I_LITE -REQ-405004/B-Transit to Failure State if the V2I_LITE System Service Stays in Standby State too Long

If the V2I_LITE_System Service stays in Standby State too long, it shall transit to the Failure state. Details of the transition are shown below:

3.1.4 *V2I_LITE-ACT-REQ-405005/A-Time to Stay in Standby Mode*

Activity Diagram

3.1.5 V2I LITE Top Level Feature State Machine

3.1.5.1 *V2I_LITE-STM-REQ-405002/A-V2I_LITE Top Level Feature States*

3.2 Non-Functional Requirements

3.3 Security Requirements



4 Functional Definition

4.1 V2I-LITE-FUN-REQ-410001/A-V2I Vehicle Setting

4.1.1 Use Cases

4.1.2 Requirements

- 4.1.2.1 V2I-LITE-REQ-410121/A-Get the V2I LITE Setting SDK Instance
- 4.1.2.2 V2I-LITE-REQ-410122/A-Connect to V2I LITE service
- 4.1.2.3 V2I-LITE-REQ-410123/A-Initialize the instance of V2I LITE Setting SDK
- 4.1.2.4 V2I-LITE-REQ-410124/A-Show the specific V2I LITE Setting Page
- 4.1.2.5 V2I-LITE-REQ-410125/A-Query Setting values
- 4.1.2.6 V2I-LITE-REQ-410126/A- onSettingOptionClick
- 4.1.2.7 V2I-LITE-REQ-410127/A-Request to apply V2I Enrollment
- 4.1.2.8 V2I-LITE-REQ-410128/A-Request to cancel V2I Enrollment
- 4.1.2.9 V2I-LITE-REQ-410129/A-Request to exit V2I Enrollment
- 4.1.2.10 V2I-LITE-REQ-4101210/A-Restore the default Settings
- 4.1.2.11 V2I-LITE-REQ-4101212/A-DisConnect to V2I LITE service
- 4.1.2.12 V2I-LITE-REQ-4101213/A-Apply V2I Favorite Operation
- 4.1.2.13 V2I-LITE-REQ-4101214/A-Cancel V2I Favorite Operation
- 4.1.2.14 V2I-LITE-REQ-4101215/A-Query V2I Favorite status
- 4.1.2.15 V2I-LITE-REQ-4101216/A-Query V2I Auth Date
- 4.1.2.16 V2I-LITE-REQ-4101217/A-Update V2I Auth Date
- 4.1.2.17 V2I-LITE-REQ-4101217/A-Pop up Toast
- 4.1.2.18 V2I-LITE-REQ-4101217/A-Page Change

『S0002 车路协同系统使用申请流程』页面 TnC (Terms & Conditions) 文言采用 HTML 格式进行加载显示。

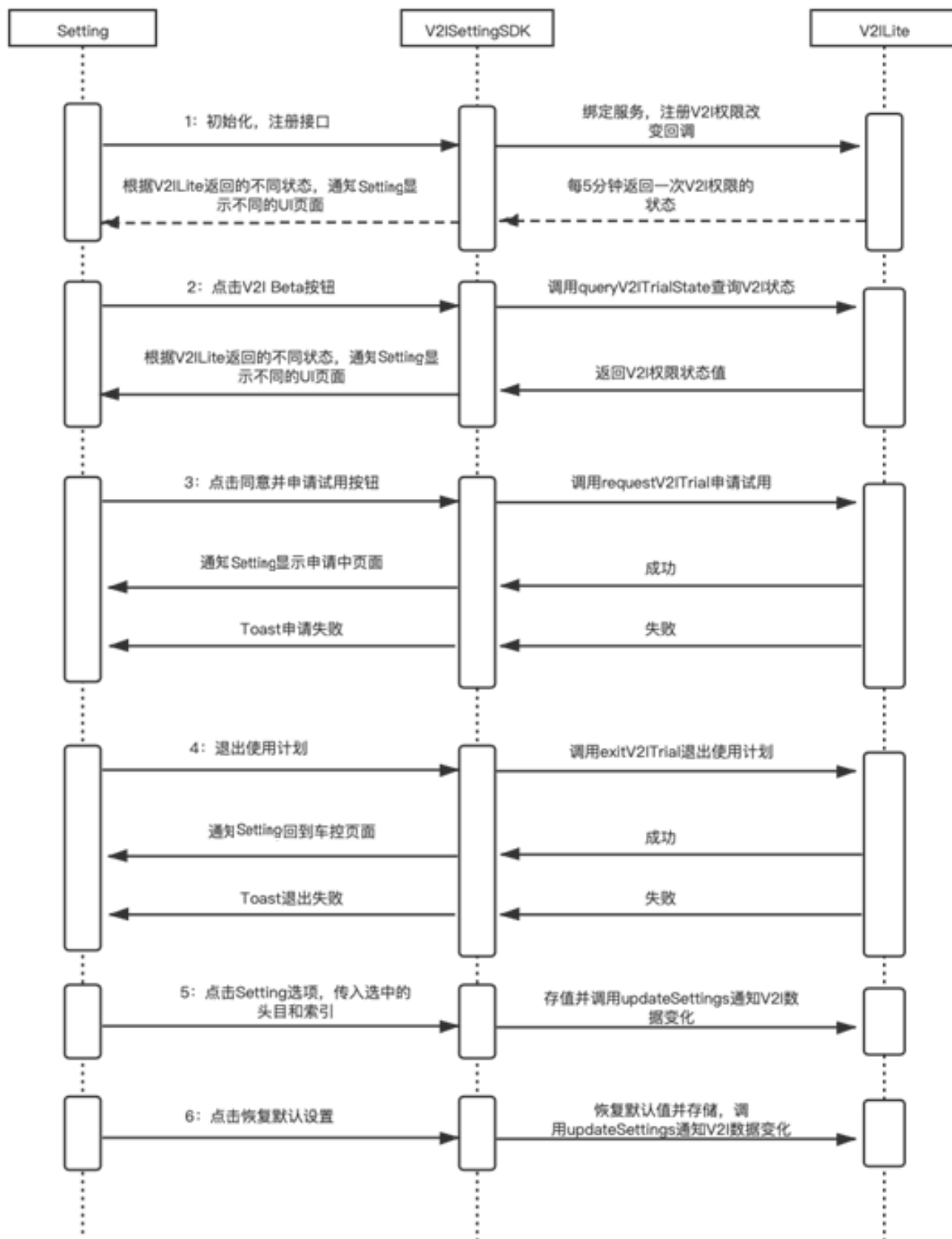


4.1.3 White Box View

4.1.3.1 *Activity Diagrams*

4.1.3.2 *Sequence Diagrams*

DRAFT





4.1.4 Interfaces & Parameters

4.1.4.1 *static V2ISettingManager getInstance(Context context)*

Field	Description
功能	获取 V2ISettingManager 实例。
参数	context Application context
返回	V2ISettingManager SDK V2ISettingManager 实例

4.1.4.2 *void connect(V2ISettingManager.ConnectionCallback callback)*

Field	Description
功能	连接 V2ILite 远程 service
参数	V2ISettingMa 用于接收结果的回调， nager.Conne 回调方法 onConnect()表明连接远程 V2ILite 服务成功 ctionCallback 回调方法 onDisconnect()表明连接远程 V2ILite 服务失败

4.1.4.3 *void init(V2ISettingListener listener)*

Field	Description
功能	初始化 SettingSDK
参数	V2ISettingLis 用于接收结果的回调， tener 通知 UI 页面变化的回调方法为 onPageChange(SettingPage settingPage)。 通知 Toast 的回调方法为 onToast(String message)。

示例代码：

```
public class YourApplication extends Application
{
    .....
    public void onCreate() {
        V2ISettingManager.getInstance(context).connect
            (new V2ISettingManager.ConnectionCallback){
            @Override
            public void onConnect() {
                V2ISettingManager.getInstance(context).init(new V2ISettingListener){
                @Override
                public void onPageChange(SettingPage settingPage) {
```



//这个回调方法中返回 Setting 需要展示的页面和对应的数据，SettingPage 是枚举类，包含了所有 Setting 需要展示的页面

```
    }  
    @Override  
    public void onToast(String message) {  
        //这个回调方法中返回 Toast 需要的 message，Setting 只需在该回调有返回的时候将信息 Toast 在屏幕上即可  
    }  
};  
  
}  
  
@Override  
public void onDisconnect() {  
  
}  
  
}
```

4.1.4.4 void onV2ISettingPageClick()

Field	Description
功能	想要展示 Setting 页面时调用，结果在 V2IListener 中返回

4.1.4.5 List<V2IActionBean> queryV2ISettingList()

Field	Description
功能	在服务城市且正常提供服务时，查询 setting 页面具体选项值时调用
返回	List<V2IActionBean> setting 页面具体选项值的 List，每个值用 V2IActionBean 类标识。当 Listener 返回“S0101 设置页面”时，需要调用此接口返回页面具体显示的 Setting 项。

4.1.4.6 void onSettingOptionClick(V2IActionBean actionBean)

Field	Description
功能	当用户点击 Setting 页面某个选项时调用，V2IActionBean 中的 Action 用来表明用户点击的具体是那个项，index 表明用户选择该项的第几条
参数	V2IActionBean 具体解释如下



V2IActionBean-Action枚举类释义

Action.ACTION_V2I_ON_OFF	允许车路协同通知
Action.ACTION_TLI_SENSITIVITY	接收红绿灯信号
Action.ACTION_GLOSA_ON_OFF	绿波引导
Action.ACTION_GLN_SENSITIVITY	绿灯起步提醒
Action.ACTION_RLVW_SENSITIVITY	闯红灯预警
Action.ACTION_RSI_ON_OFF	道路信息广播
Action.ACTION_VOICE_SETTING	声音设置
Action.ACTION_GLOBAL_OVERLAY_ON_OFF	V2I浮窗

V2IActionBean-index选值释义

开启	1
关闭	0
距离远	0
距离近	1
关闭	2
开启	0
关闭	1
灵敏度8秒	0
灵敏度5秒	1
灵敏度3秒	2
关闭	3
灵敏度高	0
灵敏度低	1
关闭	2
开启	0
关闭	1
详细	0
简洁	1
关闭	2
开启	0
关闭	1

4.1.4.7 void requestV2ITrial(string date)

Field	Description
功能	用户在申请页面点击同意并申请试用按钮时调用，结果在 V2IListener 中返回
参数	date 授权过期日期，UTC 时间,格式 yyyyMMddHHmmss(24 小时制)

4.1.4.8 void cancelV2IRequest()

Field	Description
功能	用户在申请中页面点击取消申请使用时调用，结果在 V2IListener 中返回

4.1.4.9 void exitV2ITrial()

Field	Description
功能	用户退出试用计划时调用，结果在 V2IListener 中返回

4.1.4.10 *void restoreDefaultSetting()*

Field	Description
功能	用户点击恢复默认设置时调用，结果在 V2IListener 中返回

4.1.4.11 *void disconnect()*

Field	Description
功能	断开远程 V2ILite 服务时调用

4.1.4.12 *boolean applyV2IFavorite()*

Field	Description
功能	收藏接口。“V2I 页面”中点击『收藏』按钮调用。 true: 可以收藏 false: 不可收藏

4.1.4.13 *boolean cancelV2IFavorite()*

Field	Description
功能	取消收藏。“常用设置”或“V2I 页面”中点击『取消收藏』按钮时调用。 true: 可以取消收藏 false: 不可取消收藏

4.1.4.14 *void queryV2IFavoriteState()*

Field	Description
功能	查询是否收藏的状态。“常用设置”中是否显示“允许车路协同通知”时查询；“V2I 页面”中收藏按钮是否高亮时查询。 1: 已收藏 0: 未收藏

4.1.4.15 *string queryV2IAuthDate()*

Field	Description
-------	-------------



功能

授权过期日期, UTC 时间,格式 yyyyMMddHHmmss(24 小时制)

4.1.4.16 void updateV2IAuthDate (string date)

Field	Description
功能	用户在更新授权时间按钮时调用, 结果在 V2IListener 中返回
参数	date 授权过期日期, UTC 时间,格式 yyyyMMddHHmmss(24 小时制)

4.1.4.17 void onToast(string message)

接口名	作用	返回成功	返回失败
queryV2ITrialState	获取V2I试用状态	获取使用状态成功 (不需要弹 toast)	当前功能不可用, 请稍后重试 (toast)
requestV2ITrial	申请试用权限	提交成功 (不需要弹 toast) 备注: 成功后页面会跳转至“S0004-1 申请中”页面, 无需单独提醒	车辆信息异常, 申请失败 (toast)
cancelV2IRequest	取消申请试用	取消申请成功 (不需要弹 toast) 备注: 成功后页面会跳转至“S0008 退出成功”页面, 无需单独提醒	取消申请失败 (toast)
exitV2ITrial	退出试用计划	退出使用计划成功 (不需要弹 toast) 备注: 成功后页面会跳转至“S0008 退出成功”页面, 无需单独提醒	退出使用计划失败 (toast)
updateV2IAuthDate	更新授权时间	更新授权时间成功 (不需要弹 toast) 备注: 成功后会显示更新时间, 无需单独提醒	更新失败, 请稍后重试 (toast)

注: toast 方式保持统一: 返回成功有页面跳转, 不需要弹 toast; 返回失败, 页面没有反应, 需要弹 toast。

4.1.4.18 void onPageChange(SettingPage settingPage)

Field	Description
功能	通知 UI 页面变化的回调方法。

V2I SettingListener.onPageChange()返回的枚举类 SettingPage 释义



枚举常量	页面说明
SettingPage.Function_NOT_OPEN	S0000-1 服务初始化
SettingPage.NOT_SERVING_CITY	S0007 无法使用（通过申请）
SettingPage.REQUEST_UNDER_REVIEW	S0004-1 申请中（取消申请按钮置灰不可用）
SettingPage.REQUEST_UNDER_REVIEW_2	S0004-2 申请中（取消申请按钮置灰可点击）
SettingPage.REQUEST_V2I_TRAIL	S0002 车路协同系统使用申请流程
SettingPage.SETTING_SELECT_PAGE	S0101 设置页面
SettingPage.SHOW_OTHER_PAGE	S0001-1 车辆控制-常用设置页面

枚举常量和说明**FUNCTION_NOT_OPEN**

S0000-1 服务初始化

NOT_SERVING_CITY

S0007 无法使用（通过申请）

REQUEST_NOT_PASS

审核不通过

REQUEST_UNDER_REVIEW

S0004-1 申请中（取消申请试用按钮置灰不可点击）

REQUEST_UNDER_REVIEW_2

S0004-2 申请中（取消申请试用按钮可点击）

REQUEST_V2I_TRAIL

S0002 车路协同系统使用申请流程

SETTING_SELECT_PAGE

S0101 设置页面

SHOW_OTHER_PAGE

S0001-1 车辆控制-常用设置页面

4.2 V2I-LITE-FUN-REQ-420001/A-Audio Channel Arbitration

4.2.1 Use Cases

使用定制的音频仲裁对象进行音频焦点的获取和放弃，配合原生的 mediaplayer 进行音频文件的播放。



4.2.2 Requirements

4.2.2.1 V2I-LITE-REQ-420221/A-Get the instance of MediaPlayerManager

4.2.2.2 V2I-LITE-REQ-420222/A-Request the audio focus

4.2.2.3 V2I-LITE-REQ-420223/A-Abandon the audio focus

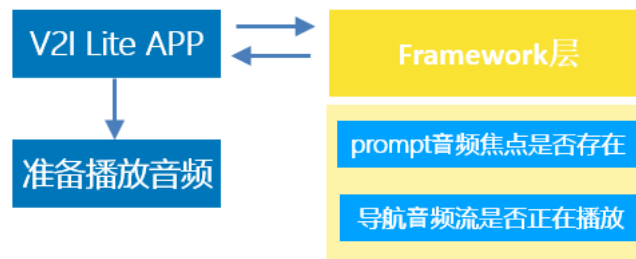
4.2.2.4 V2I-LITE-REQ-420224/A-The AudioFocusRequest

4.2.2.5 V2I-LITE-REQ-420225/A-The AudioFocusInfo

4.2.2.6 V2I-LITE-REQ-420226/A-The callback of FordCarAudioManager

4.2.3 White Box View

4.2.3.1 Activity Diagrams



Current	Media/Radio	Telephone	Prompt - Navi.User	Prompt & V2I	PTT & VR	Mute	Standby
New							
Media/Radio	Granted	Delay	Mix	Mix	Reject	Granted	Reject
Telephone	Granted	Granted 注1	Mix 注13	Granted	Granted	Granted	Reject
Prompt - Navi.User	Mix 注12	Mix	Granted	Granted	Granted	Granted	Reject
Prompt & V2I	Mix	Reject	Reject	Granted	Reject	Granted	Reject
PTT & VR	Granted	Reject	Granted	Granted	Granted 注2	Granted	Reject
Mute	mute	Reject	Reject	Reject	Reject	Granted 注5	Reject
Standby	Granted	Granted	Granted	Granted	Granted	Granted 注10	Granted

注：Phase5 与 Phase4 保持一致，不做修改；待数据质量提高后，TBD。

4.2.3.2 Sequence Diagrams

4.2.4 Interfaces & Parameters

4.2.4.1 *FordCarAudioManager (FordCarAudioManager) Car.getCarManager(SERVICE)*

Field	Description
功能	获取 FordCarAudioManager 的实例对象。
参数	SERVICE 服务类型 (Car.FORD_AUDIO_SERVICE)



返回	FordCarAudioManager 实例对象。
注意	<p>FordCarAudioManager 需要系统签名 + android:sharedUserId="android.uid.system", 如果应用需要调用音量、音效等设置接口, 还需要在 AndroidManifest 中增加如下权限:</p> <pre><uses-permission android:name="android.car.permission.CAR_CONTROL_AUDIO_SETTINGS" /> <uses-permission android:name="android.car.permission.CAR_CONTROL_AUDIO_VOLUME" /></pre>
使用	<pre>//申明 FordCarAudioManager mFordCarAudioManager; Car car; //连接&初始化 void initCar(){ car = Car.createCar(this, new ServiceConnection() { @Override public void onServiceConnected(ComponentName name, IBinder service) { try { mFordCarAudioManager = (FordCarAudioManager) car.getCarManager(Car.FORD_AUDIO_SERVICE); } catch (CarNotConnectedException e) { e.printStackTrace(); } } }); car.connect(); }</pre>

4.2.4.2 *int requestAudioFocus(FordCarAudioManager.AudioFocusRequest request)*

Field	Description
功能	请求获取音频焦点。
参数	requestAudioFocus 音频焦点对象
返回	Int 焦点获取结果
注意	需要权限: android.car.Car#PERMISSION_CAR_CONTROL_AUDIO_SETTINGS permission.



4.2.4.3 *int abandonAudioFocus(FordCarAudioManager.AudioFocusRequest request)*

Field	Description
功能	放弃音频焦点。
参数	requestAudioFocus 音频焦点对象
返回	Int 焦点放弃结果
注意	需要权限： android.car.Car#PERMISSION_CAR_CONTROL_AUDIO_SETTINGS permission.

4.2.4.4 *AudioFocusInfoClass FordCarAudioManager.AudioFocusRequest*

Field	Description
功能	焦点请求操作对象
返回	AudioFocusRequest 焦点请求实例
使用	AudioFocusRequest(AudioFocusInfo focusInfo, FordCarAudioManager.AudioFocusListener listener)

4.2.4.5 *Class AudioFocusInfo*

Field	Description
功能	对音频焦点的请求进行描述
返回	AudioFocusInfo 音频焦点描述实例
注意	音源与 USAGE 对应关系 应用音源 定制 AudioUsage 原生 AudioAttributes 媒体 AUDIO_USAGE_MEDIA AudioAttributes.USAGE_MEDIA 来电铃声/通话 AUDIO_USAGE_BT_PHONE AudioAttributes.USAGE_VOICE_COMMUNICATION 蓝牙音乐 AUDIO_USAGE_BT_MUSIC AudioAttributes.USAGE_VOICE_COMMUNICATION_SIGNALLING 蓝牙耳机 AUDIO_USAGE_BT_HEADSET AudioAttributes.USAGE_ASSISTANCE_ACCESSIBILITY VR AUDIO_USAGE_VR AudioAttributes.USAGE_ASSISTANT 导航 AUDIO_USAGE_NAVIGATION AudioAttributes.USAGE_ASSISTANCE_NAVIGATION_GUIDANCE



	紧急电话	AUDIO_USAGE_ECALL	NA
	提示/车机铃声	AUDIO_USAGE_PROMPT	AudioAttributes.USAGE_NOTIFICATION
	雷达/倒车	AUDIO_USAGE_RADAR	NA
	本地电台	AUDIO_USAGE_RADIO	NA
使用	//媒体源 AudioFocusInfo focus = new AudioFocusInfo(context, 0, AudioUsage.AUDIO_USAGE_MEDIA);		

4.2.4.6 Interface FordCarAudioManager.AudioFocusListener

Field	Description
功能	焦点请求监听回调
接口	Void dispatchAudioFocusChange(int changed)

4.3 V2I-LITE-FUN-REQ-430001/A-Text-to-Speech

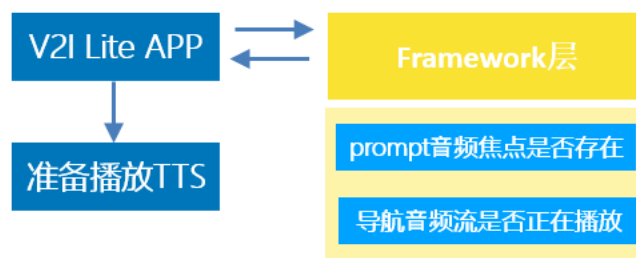
4.3.1 Use cases

4.3.2 Requirements

- 4.3.2.1 V2I-LITE-REQ-430321/ TTSPlayerManager getInstance(Context context)
- 4.3.2.2 V2I-LITE-REQ-430322/ void playTTSInPrompt(String text, ITtsListener listener)
- 4.3.2.3 V2I-LITE-REQ-430323/ void ITtsListener.onSpeechStart(int status, String message)
- 4.3.2.4 V2I-LITE-REQ-430324/ void ITtsListener.onSpeechFinish(int status, String message)
- 4.3.2.5 V2I-LITE-REQ-430325/ void ITtsListener.onError(int status, String message)
- 4.3.2.6 V2I-LITE-REQ-430326/ void ITtsListener.onSpeechStop(int status, String message)

4.3.3 White Box View

4.3.3.1 Activity Diagrams





4.3.3.2 Sequence Diagrams

4.3.4 Interfaces & Parameters

4.3.4.1 *TTSPlayerManager getInstance(Context context)*

Field	Description
功能	获取 TTSPlayerManager 的单例对象
参数	Context 上下文关系
返回	TTSPlayerManager 单例对象

4.3.4.2 *void playTTSInPrompt(String text, ITtsListener listener)*

Field	Description
功能	调用此接口在 “Prompt & V2I” 通道播放 TTS。
参数	text 需要播放的 TTS 文本
	listener 播放状态回调接口，具体需要实现的方法参考 interface 3/4/5

4.3.4.3 *void ITtsListener.onSpeechStart(int status, String message)*

Field	Description
功能	此方法用于开始播放 TTS 的回调。
参数	status 成功返回状态值，如有。
	message 播放状态回调接口，如有。

4.3.4.4 *void ITtsListener.onSpeechFinish(int status, String message)*

Field	Description
功能	此方法用于播放 TTS 结束的回调。
参数	status 成功返回状态值，如有。
	message 播放状态回调接口，如有。

4.3.4.5 *void ITtsListener.onError(int status, String message)*

Field	Description
功能	此方法用于播放 TTS 失败的回调。
参数	status 成功返回状态值，必填。
	message 播放状态回调接口，必填。



4.3.4.6 `void ITtsListener.onStop(int status, String message)`

Field	Description
功能	此方法用于播放 TTS 停止的回调。
参数	status 成功返回状态值，必填。
	message 播放状态回调接口，必填。

4.4 V2I-LITE-FUN-REQ-440001/A-CAN Signal Receiver

V2I LITE 通过 Car Service 获取 CAN 信号。

4.4.1 Use cases

4.4.2 Requirements

4.4.2.1 V2I-LITE-REQ-440421/ CarSensorManager CarSensorManager.getInstance(context)

4.4.2.2 V2I-LITE-REQ-440422/ void init(ISensorChangeListener listener)

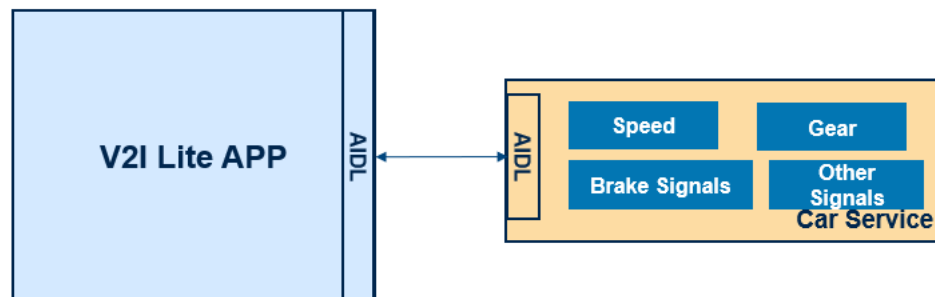
4.4.2.3 V2I-LITE-REQ-440423/ void SensorChangeListener.onSensorChange

4.4.2.4 V2I-LITE-REQ-440424/ String getVinCode()

4.4.2.5 V2I-LITE-REQ-440425/ String getXXX()

4.4.3 White Box View

4.4.3.1 Activity Diagrams



4.4.3.2 Sequence Diagrams

4.4.4 Interfaces & Parameters

4.4.4.1 `CarSensorManager CarSensorManager.getInstance(context)`

Field	Description
FILE:PHASE6_V2I_LITE_SPSS	FORD MOTOR COMPANY CONFIDENTIAL The information contained in this document is Proprietary to Ford Motor Company.
	Page 25 of 54



功能	获取 CarSensorManager 单例对象	
参数	Context	上下文关系
返回	CarSensorManager	单例对象

4.4.4.2 void init(ISensorChangeListener listener)

Field	Description
功能	用于初始化回调接口。
参数	listener 注册回调 listener，具体参考 interface 4.4.4.3.

4.4.4.3 void SensorChangeListener.onSensorChange(int eventType, int[] ints, float[] floats)

Field	Description
功能	用于接受 CAN 信号发生变化时可以接受到回调
参数	eventType eventType 具体需求参考 “Table 1: Can Signals of V2I Requirement”
	ints 根据 eventType 类型，如果为 int 数据使用此参数接受
	floats 根据 eventType 类型，如果为 float 数据使用此参数接受

4.4.4.4 String getVinCode()

Field	Description
功能	此方法用于获取当前车辆的 VIN 码。
返回	Vin 返回当前车辆的唯一识别码 VIN

4.4.4.5 String getXXX()

Field	Description
功能	此类方法用于获取各 CAN 信号的状态，需要按照信号名称单独定义方法
返回	int / float 单次获取车辆状态（车速、左转、右转等），如下

此方法用于获取各 CAN 信号的状态，具体接口名称参考如下：

Signal	ECU Tx	ECU Rx
VehicleGGCCData	BCM	APIM
Veh_V_ActlEng	ECM,PCM	GWM
TurnLghtLeft_D_Rq	BCM	APIM
TurnLghtRight_D_Rq	BCM	APIM
GearLvrPos_D_Actl	PCM,SOBDMC,TCM	APIM
BpedDrvAppl_D_Actl	ECM,PCM	APIM
StopStrtDrvMde_D_Indic	ECM,PCM	APIM

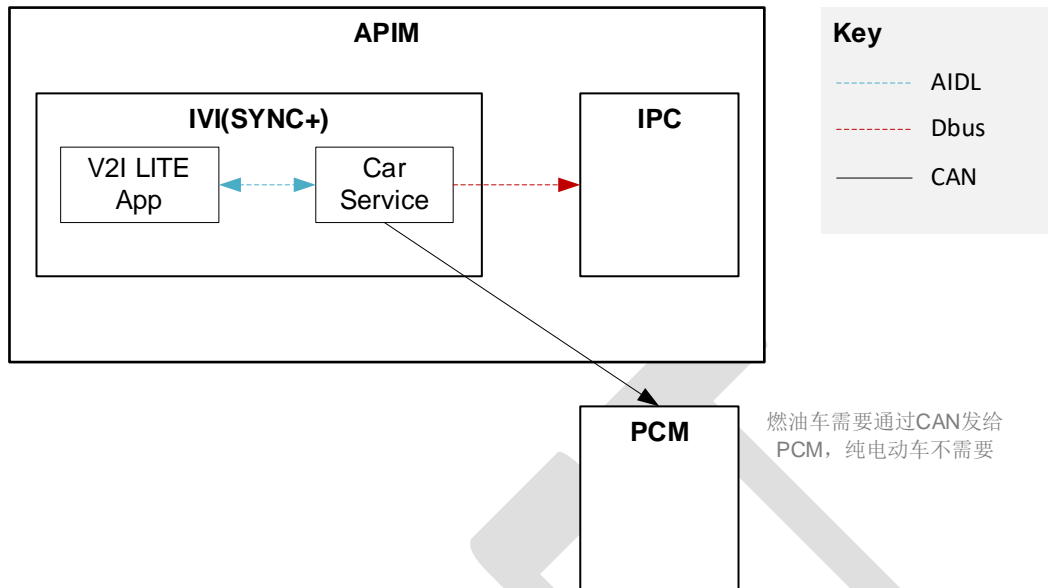


StopStrtStdby_D_Indic	ECM,PCM	APIM
StePinComp_An_Est	GFM,PSCM,SASM,SCCM	APIM
StePinCompAnEst_D_Qf	GFM,PSCM,SASM,SCCM	APIM
ApedPos_Pc_ActlArb	ECM,PCM	APIM
ApedPosPcActl_D_Qf	ECM,PCM	APIM
WhlRotatRI_No_Cnt	ABS	APIM
WhlRotatRr_No_Cnt	ABS	APIM
SteWhlComp_An_Est	ABS	APIM
SteWhlCompAnEst_D_Qf	ABS	APIM
HazrdLght_B_Stat	BCM	APIM
TurnLghtLeftOn_B_Stat	BCM	APIM
TurnLghtRightOn_B_Stat	BCM	APIM
LaActvStats_D2_Dsply	IPMA	APIM
StabCtlBrkActv_B_Actl	ABS,GFM	APIM
DrvSlipCtlLamp_D_Rq	ABS,GFM	APIM
DrvAntiLckLamp_D_Rq	ABS,GFM	APIM
DrvSlipCtlOffLamp_D_Rq	ABS,GFM	APIM

4.5 V2I-LITE-FUN-REQ-450001/A-CAN/Dbus Signal Transmitter

V2I LITE 一般通过 Car Service 进行信号发送。具体需求如下：

1. 需要把 V2I 相关的数据发送给仪表，让仪表显示 V2I 信息。
 - a. 如果是 CDC (Central Display Controller) 架构，IPC 和 IVI 之间通过 Dbus 通信。
 - b. 如果不是 CDC 架构，则通过 CAN 通信，在传送过程中，应保证数据送达，并要求延迟在 100ms 以内。详细指标可参考 CAN 网络数据传输要求。
2. 根据车型配置情况，需要把 V2I 数据发送给 PCM，以实现其相关功能。
 - a. 如果是车型为燃油车，需要同时把 V2I 信号通过 CAN 转发给 PCM；
 - b. 如果是纯电动，则不需要转发给 PCM。
3. 根据车型配置情况，对于配置 HUD 的车型，需要把 V2I 数据发送给 HUD，让 HUD 显示 V2I 信息。
 - a. 如果 HUD 与 APIM 通过 CAN 通信，则需要 Car Service 转发到 CAN 上；
 - b. 如果 HUD 软件部署在 IVI 上，则 HUD APP 可通过 AIDL 与 Car Service 进行通信，获取 V2I 数据进行显示。



4.5.1 Use cases

4.5.2 Requirements

- 4.5.2.1 V2I-LITE-REQ-450521/ static FordV2ILiteManager getInstance
- 4.5.2.2 V2I-LITE-REQ-450522/ void onConnect(ConnectionCallback callback)
- 4.5.2.3 V2I-LITE-REQ-450523/ void sendTrafficLight Data 1(byte[] data)
- 4.5.2.4 V2I-LITE-REQ-450524/ void sendTrafficLight Data 2(byte[] data)
- 4.5.2.5 V2I-LITE-REQ-450525/ Interface ConnectionCallback

4.5.3 White Box View

- 4.5.3.1 **Activity Diagrams**
- 4.5.3.2 **Sequence Diagrams**

4.5.4 Interfaces & Parameters

- 4.5.4.1 static FordV2ILiteManager getInstance(Context context)

Field		Description
功能	获取 FordV2ILiteManager 实例。	
参数	context	Application context
返回	FordManager	FordV2ILiteManager 实例，使用此实例进行后续接口的调用。

4.5.4.2 *void onConnect(ConnectionCallback callback)*

Field		Description
功能	向 Ford-SDK 提供状态回调接口，Ford-SDK 利用此接口把初始化工作的状态通知到 V2ILite	
参数	context	Application context
	callback	Ford-SDK 的初始化状态通过调用此接口通知到 V2ILite，参照...4.5 关于此接口的具体说明

4.5.4.3 *void sendTrafficLight_Data_1(byte[] data)*

Field		Description
功能	向 Can 总线发送 data 数据，Message ID 是 0X273	
参数	data	待发送的 byte 数组数据,数据的格式要求如下表所示

表 0x273 对应数据格式

Message ID	Length	Byte[x]	Byte[x]'s Name	Byte[x]'s Description (MSB)	Definition in Source Code	Needed by Cluster
0	8	Byte[0]-Byte[7]	RoadSignInfo1_D_Dsply	范围：0-255		Yes
	16	Byte[16]-Byte[31]	RoadStopLine_L_Est	范围：0-65533		No
	1	Byte[32]	TrfcLghtVActv_B_Stat	范围：0-1		Yes
	3	Byte[33]-Byte[35]	TrfcLghtDir_D_Stat	范围：0-7		Yes
	3	Byte[36]-Byte[38]	TrfcLght_D_Stat	范围：0-7		Yes
	1	Byte[39]	RoadStopLineL_B_Avail	范围：0-1		No
	8	Byte[40]-Byte[47]	TrfcLght_V_Mn	范围：0-253		Yes
	8	Byte[48]-Byte[55]	TrfcLght_V_Mx	范围：0-253		Yes
	1	Byte[56]	TrfcLghtWarnActv_B_Stat	范围：0-1		Yes
	7	Byte[57]-Byte[63]	TrfcLghtDecel_A_Dsply	范围：0-125		No



Message ID	Length	Byte[x]	Byte[x]'s Name	Byte[x]'s Description (MSB)	Definition in Source Code	Needed by Cluster
	3	Byte[64]- Byte[66]	VehToInfrstrctr_D_Stat	范围: 0-7		Yes

4.5.4.4 void sendTrafficLight_Data_2(byte[] data)

Field		Description
功能	向 Can 总线发送 data 数据, Message ID 是 0X276	
参数	data	待发送的 byte 数组数据,数据的格式要求如下表所示

表 0x276 对应数据格式

Message ID	Length	Byte[x]	Byte[x]'s Name	Byte[x]'s Description (MSB)	Definition in Source Code	Needed by Cluster
1	8	Byte[0]- Byte[7]	TrfcLghtLeft_T_Actl	范围: 0-253		Yes
	2	Byte[8]- Byte[9]	TrfcLghtLeftType_D_Stat	范围: 0-3		Yes
	3	Byte[10]- Byte[12]	TrfcLghtLeftOn_D_Stat	范围: 0-7		Yes
	3	Byte[13]- Byte[15]	TrfcLghtLeftColr_D_Stat	范围: 0-7		Yes
	8	Byte[16]- Byte[23]	TrfcLghtRight_T_Actl	范围: 0-253		Yes
	2	Byte[24]- Byte[25]	TrfcLghtRightType_D_St	范围: 0-3		Yes
	3	Byte[26]- Byte[28]	TrfcLghtRightOn_D_Stat	范围: 0-7		Yes
	3	Byte[29]- Byte[31]	TrfcLghtRightColr_D_St	范围: 0-7		Yes
	8	Byte[32]- Byte[39]	TrfcLghtStrght_T_Actl	范围: 0-253		Yes



Message ID	Length	Byte[x]	Byte[x]'s Name	Byte[x]'s Description (MSB)	Definition in Source Code	Needed by Cluster
	2	Byte[40]-Byte[41]	TrfcLghtStrghtType_D_St	范围：0-3		Yes
	3	Byte[42]-Byte[44]	TrfcLghtStrghtOn_D_Stat	范围：0-7		Yes
	3	Byte[45]-Byte[47]	TrfcLghtStrghtColr_D_St	范围：0-7		Yes
	8	Byte[48]-Byte[55]	TrfcLghtUturn_T_Actl	范围：0-253		Yes
	2	Byte[56]-Byte[57]	TrfcLghtUturnType_D_St	范围：0-3		Yes
	3	Byte[58]-Byte[60]	TrfcLghtUturnOn_D_Stat	范围：0-7		Yes
	3	Byte[61]-Byte[63]	TrfcLghtUturnColr_D_St	范围：0-7		Yes

4.5.4.5 Interface ConnectionCallback

Field	Description
功能	初始化状态通知到 V2I Lite
参数	onSuccess
	onFail

4.6 V2I-LITE-FUN-REQ-460001/A-Keystore Encryption & Decryption

4.6.1 Use cases

Phase2/3: 通过原生方式实现与城市平台进行加解密操作。

Phase4: 通过使用 KeyStore 方式实现加解密操作。保证 1s 内可以成功完成 24 次数据的加解密。

Phase5: 通过使用 TEE 完成针对不同城市平台的加解密。详见 V2I-LITE-FUN-REQ-4180001/TEE



4.6.2 Requirements

4.6.2.1 V2I-LITE-REQ-460621/ KeyStoreManager getInstance

4.6.2.2 V2I-LITE-REQ-460622/ void updateKey

4.6.2.3 V2I-LITE-REQ-460623/ String encryptData

4.6.2.4 V2I-LITE-REQ-460624/ String decryptData

4.6.3 White Box View

4.6.3.1 **Activity Diagrams**

4.6.3.2 **Sequence Diagrams**

4.6.4 Interfaces & Parameters

4.6.4.1 *KeyStoreManager getInstance*

Field	Object Type		Description
功能	通过此接口可以获取 KeyStoreManager 单例对象		
参数	-	-	-
返回值	-	KeyStoreManager	返回 KeyStoreManager 实例对象用于后续加解密操作。

4.6.4.2 *void updateKey(String key, Date expired, String alias, AES_KEY_TYPE keyType)*

Field	Object Type		Description
功能	通过此接口可以添加或更新 key 信息		
参数	key	String	key 变量
	expired	Date	有效期限
	alias	String	对应 key 别名
	keyType	AES_KEY_TYPE	转化类型，如 AES/CBC/PKCS5Padding/AES/ECB/PKCS5Padding
返回值	-	-	-

4.6.4.3 *API03- String encryptData(byte[] toEncryptData, String alias, AES_KEY_TYPE keyType, String iv)*

Field	Object Type		Description
功能	通过此接口可以将数据进行加密。		



参数	toEncryptData	byte[]	加密前的原始数据
	alias	String	对应 key 别名
	keyType	AES_KEY_TYPE	转化类型，如 AES/CBC/PKCS5Padding/AES/ECB/PKCS5Padding
	iv	String	偏移量，如 IvParameterSpec 中 iv
返回值	-	String	加密成功后的数据

4.6.4.4 API04- String decryptData(byte[] toDecryptData, String alias, AES_KEY_TYPE keyType, String iv)

Field	Object Type		Description
功能	通过此接口可以将数据进行解密。		
参数	toDecryptData	byte[]	解密前的原始数据
	alias	String	对应 key 别名
	keyType	AES_KEY_TYPE	转化类型，如 AES/CBC/PKCS5Padding/AES/ECB/PKCS5Padding
	iv	String	偏移量，如 IvParameterSpec 中 iv
返回值	-	String	解密成功后的数据



4.7 V2I-LITE-FUN-REQ-470001/A-Embedded Navigation

4.7.1 Use cases

4.7.2 Requirements

- 4.7.2.1 V2I-LITE-REQ-470721/ static V2ISDKProxy getInstance(Context context)
- 4.7.2.2 V2I-LITE-REQ-470722/ void setup(Context context, V2IListener<Object> listener)
- 4.7.2.3 V2I-LITE-REQ-470723/ void getCarPosition(V2IListener<V2IPosition> listener)
- 4.7.2.4 V2I-LITE-REQ-470724/ void getCityInfo(V2IListener<V2IPosition> listener)
- 4.7.2.5 V2I-LITE-REQ-470725/ void requestCrossingData(double lng, double lat, V2IListener<V2IPosition> listener)
- 4.7.2.6 V2I-LITE-REQ-470726 CarHeadingInfo getCarHeadingInfo()
- 4.7.2.7 V2I-LITE-REQ-470727/ Void setMapStatusListener(MapStatusCallback callback)
- 4.7.2.8 V2I-LITE-REQ-470728/ void sendTrafficLightRSIInfo(Bitmap highLightView, Bitmap bubbleView)
- 4.7.2.9 V2I-LITE-REQ-470729/ Double getRSIWidth()
- 4.7.2.10 V2I-LITE-REQ-4707210/ Interface V2IListener<Object>
- 4.7.2.11 V2I-LITE-REQ-4707211/ Interface V2IListener<V2IPosition>
- 4.7.2.12 V2I-LITE-REQ-4707212/ Interface V2IListener<V2IPosition>
- 4.7.2.13 V2I-LITE-REQ-4707213/ Interface V2IListener<String>
- 4.7.2.14 V2I-LITE-REQ-4707214/ CarHeadingInfo
- 4.7.2.15 V2I-LITE-REQ-4707215/ V2IPosition
- 4.7.2.16 V2I-LITE-REQ-4707216/ Interface MapstatusCallback

4.7.3 White Box View

4.7.3.1 Activity Diagrams

4.7.3.2 Sequence Diagrams

4.7.4 Interfaces & Parameters

4.7.4.1 static V2ISDKProxy getInstance(Context context)

Field	Description
功能	获取 V2ISDKProxy 实例。



参数	context	Application context
返回	V2ISDKProxy	V2ISDKProxy 实例，使用此实例进行后续接口的调用。

4.7.4.2 *void setup(Context context, V2IListener<Object> listener)*

Field	Description	
功能	向 MAP-SDK 提供状态回调接口，MAP-SDK 利用此接口把初始化工作的状态通知到 V2ILite	
参数	context	Application context
	listener	MAP-SDK 的初始化状态通过调用此接口通知到 V2ILite，参照 3.10 关于此接口的具体说明

4.7.4.3 *void getCarPosition(V2IListener<V2IPosition> listener)*

Field	Description	
功能	获取车辆所在地理位置，并且通过回调方式通知到 V2ILite	
参数	listener	通过此回调接口获得 MAP-SDK 提供的地理位置,参照 3.10 关于此接口的具体说明

4.7.4.4 *void getCityInfo(V2IListener<V2IPosition> listener)*

Field	Description	
功能	获取车辆所处的城市信息，包含了城市 ID&城市名字	
参数	listener	通过此回调接口获得 MAP-SDK 提供的城市信息.参照 3.11 关于此接口的具体说明

4.7.4.5 *void requestCrossingData(double lng,double lat, V2IListener<V2IPosition> listener)*

Field	Description	
功能	利用此接口获取路网数据，并通过回调的方式通知到 V2ILite	
参数	lng	路口经度单位 GCJ-02
	lat	路口纬度单位 GCJ-02
	listener	通过此回调 MAP-SDK 将路网数据返回给 V2ILite，此回调的具体描述参照 3.12 关于此接口的定义

4.7.4.6 *CarHeadingInfo getCarHeadingInfo()*

Field	Description
-------	-------------



功能	获取车辆导航相关信息包括道路类型、导航方向、机动点位置信息等信息	
参数		
返回	CarHeadingInfo	此类型中包含了导航相关的信息，包括道路类型、导航方向、机动点位置信息等信息。关于 CarHeadingInfo 的描述参照 3.13

4.7.4.7 *Void setMapStatusListener(MapStatusCallback callback)*

Field	Description	
功能	向 MAP-SDK 注册回调接口，MAP-SDK 通过此接口将不同的地图状态通知到 V2ILite，地图状态包括地图在前台、地图不在前台、导航中状态、巡航中状态等	
参数	callback	用于接收地图状态结果的回调。

4.7.4.8 *void sendTrafficLightRSIInfo(Bitmap highLightView, Bitmap bubbleView)*

Field	Description	
功能	通过此接口 V2ILite 将需要在地图上显示的信息发送给 MAP-SDK，MAP-SDK 将对应的信息展示在地图上。	
参数	highLightView	需要显示的交通灯 ImageView
	bubbleView	需要显示的 RSI ImageView

4.7.4.9 *Double getRSIWidth()*

Field	Description	
功能	通过此接口获取百度地图上显示的 RSI 的宽度	
参数		
返回值		在特定车型上的百度地图显示 RSI 等图像的宽度，单位是像素。



4.7.4.10 *Interface V2IListener<Object>*

```
public interface V2IListener<Object> {  
    int ERROR_NOT_SETUP = 101;  
    int ERROR_SERVICE_DISCONNECT = 102;  
    int ERROR_SERVICE_CONNECT_FAIL = 103;  
    int ERROR_SERVICE_DIED = 104;  
    int ERROR_PARAMETER_INVALID = 105;  
    int ERROR_RESPONSE_DATA_ERROR = 106;  
    int ERROR_REMOTE_ERROR = 107;  
    int ERROR_HTTP_ERROR = 108;  
    //MAP-SDK 初始化成功通过此接口回调给 V2ILite  
    void onSuccess(Object var1);  
    //MAP-SDK 初始化失败通过此接口回调给 V2ILite,var1 为错误码, var2 为错误描述信息  
    void onError(int var1, String var2);  
}
```

4.7.4.11 *Interface V2IListener<V2IPosition>*

```
public interface V2IListener<V2IPosition> {  
    int ERROR_NOT_SETUP = 101;  
    int ERROR_SERVICE_DISCONNECT = 102;  
    int ERROR_SERVICE_CONNECT_FAIL = 103;  
    int ERROR_SERVICE_DIED = 104;  
    int ERROR_PARAMETER_INVALID = 105;  
    int ERROR_RESPONSE_DATA_ERROR = 106;  
    int ERROR_REMOTE_ERROR = 107;  
    int ERROR_HTTP_ERROR = 108;  
    //MAP-SDK 通过此方法把成功获取的车辆经纬度信息回调给 V2ILite  
    void onSuccess(V2IPosition var1);  
    //MAP-SDK 通过此方法把未能成功获取车辆经纬度信息的错误代码&错误描述回调给//V2ILite  
    void onError(int var1, String var2);  
}
```

4.7.4.12 *Interface V2IListener<V2IPosition>*

```
public interface V2IListener<V2IPosition> {  
    int ERROR_NOT_SETUP = 101;  
    int ERROR_SERVICE_DISCONNECT = 102;  
    int ERROR_SERVICE_CONNECT_FAIL = 103;  
    int ERROR_SERVICE_DIED = 104;  
    int ERROR_PARAMETER_INVALID = 105;  
    int ERROR_RESPONSE_DATA_ERROR = 106;  
    int ERROR_REMOTE_ERROR = 107;  
    int ERROR_HTTP_ERROR = 108;  
    //MAP-SDK 通过此方法把成功获取的城市信息回调给 V2ILite  
    void onSuccess(V2IPosition var1);  
    //MAP-SDK 通过此方法把未能成功获取城市信息的错误代码&错误描述回调给//V2ILite  
    void onError(int var1, String var2);  
}
```

4.7.4.13 *Interface V2IListener<String>*

```
public interface V2IListener<String> {  
    int ERROR_NOT_SETUP = 101;  
    int ERROR_SERVICE_DISCONNECT = 102;  
    int ERROR_SERVICE_CONNECT_FAIL = 103;  
    int ERROR_SERVICE_DIED = 104;  
    int ERROR_PARAMETER_INVALID = 105;  
    int ERROR_RESPONSE_DATA_ERROR = 106;  
    int ERROR_REMOTE_ERROR = 107;  
    int ERROR_HTTP_ERROR = 108;  
    //MAP-SDK 通过此方法把成功获取的路网信息回调给 V2ILite  
    void onSuccess(String var1);  
    //MAP-SDK 通过此方法把未能成功获取城市路网信息的错误代码&错误描述回调给//V2ILite  
    void onError(int var1, String var2);  
}
```



4.7.4.14 CarHeadingInfo

```
public class V2ICarHeadingInfo implements Parcelable {  
    //导航方向  
    private int direction = -1;  
    //道路类型:  
    /**  
        *      主路 int MAIN_ROAD = 1;  
        *      辅路 int AUXILIARY_ROAD = 2;  
        *      高架上 int ON_THE_VIADUCTS = 4;  
        *      高架下 int UNDER_THE_VIADUCTS = 8;  
        *      高架上主路 int ON_THE_VIADUCTS_MAIN_ROAD = 5;  
        *      高架上辅路 int ON_THE_VIADUCTS_AUXILIARY_ROAD = 6;  
        *      高架下辅路 int UNDER_THE_VIADUCTS_AUXILIARY_ROAD = 10;  
        *      高架下主路 int UNDER_THE_VIADUCTS_MAIN_ROAD = 9;  
    */  
    private int roadType = 0;  
    //V2IPosition 的具体说明参照 3.14  
    private V2IPosition coordinate;  
}
```

4.7.4.15 *V2IPosition*

```
public class V2IPosition implements Parcelable {  
    private double longitude;  
    private double latitude;  
    private String city = "";  
    private int cityId;  
    private double bearing;  
    private double accuracy;  
    private long time;  
    private int satellitesCount;  
}
```

4.7.4.16 *Interface MapstatusCallback*

```
public interface MapstatusCallback {  
    //通过此接口返回地图的显示状态给 V2ILite，其中 0 表示地图可见，1 表示地图不可见  
    //2:进入导航，3:退出导航，4:进入巡航，5:退出巡航，6:进入 ar 导航，7:退出 ar 导航。  
    void onMapStatusChanged(int status);  
}
```

A	B		C
	status	状态	
0		地图可见	
1		地图不可见	
2		进入导航	
3		退出导航	
4		进入巡航	
5		退出巡航	
6		进入AR导航	
7		退出AR导航	

```
//AR 导航车机所处位置发生变化之后通过此接口调用通知到 V2ILite,0:表示处于屏幕左边，1:表示处于屏幕右侧  
void onARNavStatusChaned(int position)
```




4.8 V2I-LITE-FUN-REQ-480001/A- self-initialization

4.8.1 Use cases

4.8.2 Requirements

4.8.2.1 V2I-LITE-REQ-480821/ V2ILite self- initialization

4.8.3 White Box View

4.8.3.1 *Activity Diagrams*

4.8.3.2 *Sequence Diagrams*

4.8.4 Interfaces & Parameters

4.8.4.1 V2ILite 开机自启动

Field	Description
功能	需要系统提供一种方式保证 V2I APP 能够在开机后两分钟内完成启动，以 Phase 4 为例如下。 步骤一： 将 V2ILite.apk 存放在目录/system/app/V2ILite 下 步骤二： 将此路径配置到/system/etc 下的 first_apps.list 文件中，即可实现开机自启动 在 Phase5 上，YFV 建议首先尝试 Android 原生 BOOT_COMPLETED 广播，拉起应用。

4.9 V2I-LITE-FUN-REQ-490001/A-WIR Policy Table

4.9.1 Use cases

4.9.2 Requirements

4.9.2.1 V2I-LITE-REQ-490921/ WIR Policy Table

4.9.3 White Box View

4.9.3.1 *Activity Diagrams*

4.9.3.2 *Sequence Diagrams*

4.9.4 Interfaces & Parameters

4.9.4.1 WIR Policy Table

Field	Description
功能	需要 WIR 配置 V2I 的上网权限，保证 V2I 可以通过 UDP / HTTP 等方式获取到数据服务

WIR 通过包名来进行上网权限控制。

Phase 5 Package Name for WIR policy



Feature Name	Application ID / Package Name	Internet Interface	FTCP App(means if the App communicates to Ford cloud with FCI, if yes then it's FTCP App)
V2I Lite	com.ford.xiaoduos.v2x	5G Cellular, WiFi	NO

4.10 V2I-LITE-FUN-REQ-4100001/A-IP Whitelist

4.10.1 Use cases

4.10.2 Requirements

4.10.2.1 V2I-LITE-REQ-4101021/ IP 白名单

4.10.3 White Box View

4.10.3.1 *Activity Diagrams*

4.10.3.2 *Sequence Diagrams*

4.10.4 Interfaces & Parameters

4.10.4.1 防火墙白名单配置

Field	Description
功能	根据福特安全的要求, 需要将 V2I 访问的域名或 IP 端口配置到白名单中, 以保证只有在白名单中的 IP 端口或域名可以正常获取到数据。如下表。

UDP 访问城市平台通过将域名或端口号加入白名单的方式实现网络安全。

APP 名	APP 责任方	IP 地址是否固定	IP 地址	端口号	域名	TCP 连接有哪方发起	包名	签名信息	备注
v2i_lite.apk	zhang yuan yan	是	112.25.66.161	28120	无	UDP 连接, 车机端发起	com.ford.xiaoduos.v2x	platform key	无锡
		是	218.76.44.22	21000	无	UDP 连接, 车机端发起	com.ford.xiaoduos.v2x	platform key	长沙 QA
		是	218.76.44.22	21001	无	UDP 连接, 车机端发起	com.ford.xiaoduos.v2x	platform key	长沙 PRD
		否	219.239.83.74	4021	*.trafficechservices.com	UDP 连接, 车机端发起	com.ford.xiaoduos.v2x	platform key	南京 PRD
		是	218.30.180.77	21000	无	UDP 连接, 车机端发起	com.ford.xiaoduos.v2x	platform key	北京 PRD
		是	219.135.214.196	21000	无	UDP 连接, 车机端发起	com.ford.xiaoduos.v2x	platform key	广州 PRD
		是	?	21000	无	UDP 连接, 车机端发起	com.ford.xiaoduos.v2x	platform key	沧州 PRD
		是	?	21000	无	UDP 连接, 车机端发起	com.ford.xiaoduos.v2x	platform key	阳泉 PRD



APP 名	APP 责任方	IP 地址是否固定	IP 地址	端口号	域名	TCP 连接有哪方发起	包名	签名信息	备注
		-	-	-	https://vehiclestgcn.api.mps.ford.com.cn	HTTPS, 车机端发起	com.ford.xiaoduos.v2x	platform key	V2I Cloud STG
		-	-	-	https://vehiclecn.api.mps.ford.com.cn	HTTPS, 车机端发起	com.ford.xiaoduos.v2x	platform key	V2I Cloud PRD

4.11 V2I-LITE-FUN-REQ-4110001/A-IPPT

4.11.1 Use cases

4.11.2 Requirements

4.11.2.1 V2I-LITE-REQ-4111121/ FordIpptClientManager getInstance

4.11.2.2 V2I-LITE-REQ-4111122/ void connect

4.11.2.3 V2I-LITE-REQ-4111123/ void addIpptListenerListeners

4.11.2.4 V2I-LITE-REQ-4111124/ int registerClient()

4.11.2.5 V2I-LITE-REQ-4111125/ int sendTokenRequest

4.11.2.6 V2I-LITE-REQ-4111126/ int unregisterClient()

4.11.2.7 V2I-LITE-REQ-4111127/ interface ConnectionCallback

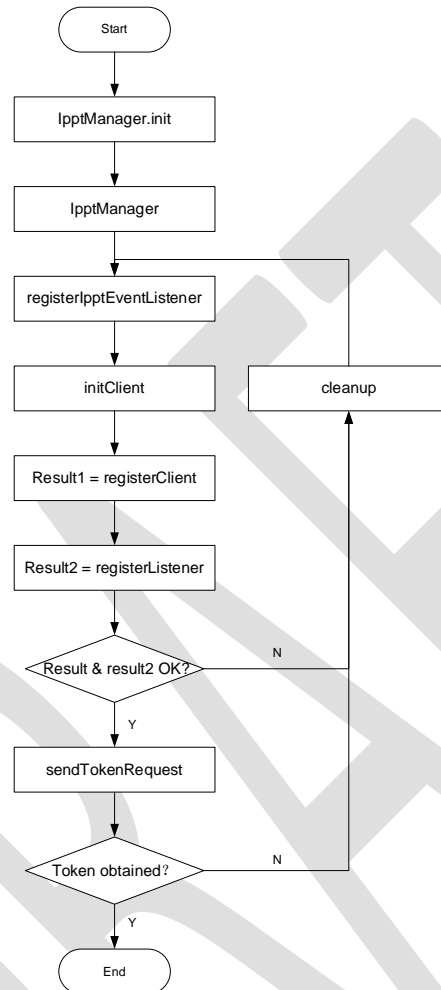
4.11.2.8 V2I-LITE-REQ-4111128/ interface IpptListener



4.11.3 White Box View

4.11.3.1 Activity Diagrams

4.11.3.2 Sequence Diagrams



The Normal token acquisition interface calling sequence is as follows:

1. **init**(Context context, ServiceConnectListener listener)
2. **registerIpptEventListener**(IIpptEventListener ipptEventListener)
3. **initClient**(String name)
4. **registerClient**() or **registerClientAsync**(), if use **registerClientAsync**, will receive callback of **OnRegisterAsyncCallback**
5. **registerListener**()
6. **sendTokenRequest**(String transactionId, Boolean hidPolicy, List<RequestConfiguration> config, List<String> scopes), after **sendTokenRequest** is called, will receive **OnTokenResponse** and callback of **OnTokenResponseStatus**
7. After received callback of **OnTokenRevocation**(String revocationId), it can be called based on the content of callback of **sendTokenRevocationResponse**, and then call **sendTokenRevocationResult**
8. At last, you can call **cleanup**() to clear the callback and unregister the client or **unregisterClient**() to unregister the client
9. Call **unregisterIpptEventListener** to logout callback



4.11.4 Interfaces & Parameters

4.11.4.1 *FordIpptClientManager getInstance*

Field	Object Type		Description
功能	通过此接口可以获取 FordIpptClientManager 单例对象		
参数	-	-	-
返回值	-	FordIpptClientManager	返回 FordIpptClientManager 实例对象用于后续操作。

4.11.4.2 *void connect(ConnectionCallback connection)*

Field	Object Type		Description
功能	通过此接口可以进行 IPPT 连接		
参数	connection	ConnectionCallback	IPPT 连接回调
返回值	-	-	-

4.11.4.3 *void addIpptListenerListeners(FordIpptClientManager.IpptListener ipptListener)*

Field	Object Type		Description
功能	通过此接口用户添加 IPPT 监听，用以获取 IPPT 状态变化。		
参数	ipptListener	IpptListener	IPPT 监听回调
返回值	-	-	-

4.11.4.4 *int registerClient()*

Field	Object Type		Description
功能	通过此接口可以进行 IPPT 客户端注册。		
参数	-	-	-
返回值	-	int	注册操作结果

**4.11.4.5** *int sendTokenRequest(String transactionId, boolean policyValidation, List configurations, List scopes)*

Field	Object Type		Description
功能	通过此接口可以进行 Token 获取请求。		
参数	transactionId	String	事物 id 序号
参数	policyValidation	boolean	
参数	configurations	List	配置信息的键值队列表
参数	scopes	List	
返回值	-	int	操作结果

4.11.4.6 *int unregisterClient()*

Field	Object Type		Description
功能	通过此接口可以进行 IPPT 客户端注销。		
参数	-	-	-
返回值	-	int	注销操作结果

4.11.4.7 *interface ConnectionCallback*

```
public interface ConnectionCallback {  
    void onConnect();  
    void onDisconnect();  
}
```

4.11.4.8 *interface IlpptListener*

```
public interface IlpptListener {  
  
    void registerCallback(int var1);  
  
    void cloudConnectivityStatusCallback(boolean var1);  
  
    void serverStatusCallback(int var1);  
  
    void tokenRequestStatusCallback(String var1, int var2, String var3);  
  
    void tokenResponseCallback(String var1, TokenResponseData var2);  
  
    void tokenResponseErrorCallback(String var1, int var2);  
  
    void tokenRevocationCallback(String var1);  
  
}
```

4.12 V2I-LITE-FUN-REQ-4120001/A-MasterReset

4.12.1 Use cases

4.12.2 Requirements

4.12.2.1 V2I-LITE-REQ-4121221/ FordMasterResetManager getInstance(Context context)4.12.2.2 V2I-LITE-REQ-4121222/ int init(IResetListener listener)4.12.2.3 V2I-LITE-REQ-4121223/ IResetListener.onChange(int status, String msg)

4.12.3 White Box View

4.12.3.1 **Activity Diagrams**4.12.3.2 **Sequence Diagrams**

4.12.4 Interfaces & Parameters

4.12.4.1 *FordMasterResetManager getInstance(Context context)*

Field	Object Type		Description
功能	通过此接口可以获取 FordMasterResetManager 单例对象		
参数	context	Context	传递 App 上下文。



返回值	masterResetManager	FordMasterResetManager	返回 FordMasterResetManager 对象用于后续初始化等操作。
-----	--------------------	------------------------	---

4.12.4.2 *int init(IResetListener listener)*

Field	Object Type		Description
功能			通过此接口可以初始化 master reset 通知服务。返回初始化的状态：“1 成功”，“0 失败”。
参数	listener	IResetListener	异步回调，状态发生变化时通过 onChange 方法通知最新的状态值。
返回值	status	int	返回初始化的状态：“1 成功”，“0 失败”。

4.12.4.3 *IResetListener.onChange(int status, String msg)*

Field	Object Type		Description
功能			通过此接口可以通知当前最新的 master reset 状态。
参数	status	int	reset 状态，至少包含的状态有 1.“复位成功” 0.“复位失败”
	msg	String	status 相对应的说明信息
返回值	-	-	-

4.13 V2I-LITE-FUN-REQ-4130001/A-SYNCP

4.13.1 Use cases

4.13.2 Requirements

4.13.2.1 V2I-LITE-REQ-4131321/ FordSyncPManager getInstance

4.13.2.2 V2I-LITE-REQ-4131322/ int initialize syncp()

4.13.2.3 V2I-LITE-REQ-4131323/ byte[] decode(byte[] decodeBuffer)



4.13.3 White Box View

4.13.3.1 Activity Diagrams

4.13.3.2 Sequence Diagrams

4.13.4 Interfaces & Parameters

4.13.4.1 FordSyncPManager getInstance(Context context)

Field	Object Type		Description
功能	通过此接口可以获取 FordSyncPManager 单例对象		
参数	context	Context	传递 App 上下文。
返回值	-	FordSyncPManager	返回 FordSyncPManager 对象用于后续初始化、解密操作。

4.13.4.2 int initialize_syncp()

Field	Object Type		Description
功能	通过此接口可以初始化 syncp 服务。返回初始化的状态：“1 成功”，“0 失败”。 备注：目前的逻辑来看初始化只有 AHU 已经通过 Provisioning 才能正常工作，所以如果 provisioning 没通过应该返回失败。		
参数	-	-	-
返回值	-	int	返回初始化的状态：“1 成功”，“0 失败”。

4.13.4.3 byte[] decode(byte[] decodeBuffer)

Field	Object Type		Description
功能	通过此接口可以将 syncp 加密过的数据进行解密。		
参数	decodeBuffer	byte[]	解密前的原始数据
返回值	-	byte[]	解密成功后的数据



4.14 V2I-LITE-FUN-REQ-4140001/A-getTheme

4.14.1 Use cases

4.14.2 Requirements

4.14.2.1 V2I-LITE-REQ-4141421/ getTheme()

4.14.2.2 V2I-LITE-REQ-4161621/ registerThemeChangedCallback(Callback callback)

4.14.3 White Box View

4.14.3.1 *Activity Diagrams*

4.14.3.2 *Sequence Diagrams*

4.14.4 Interfaces & Parameters

4.14.4.1 *getTheme()*

Field	Object Type		Description
功能	当前主题得读取 主题代表的 int :0 1 2 对应的如下: 2 是默认主题;		
参数	-	-	-
返回值	int	0-"静谧天成" 1- "静雅天成" 2- "默认主题"	

4.14.4.2 *registerThemeChangedCallback (Callback callback)*

Field	Object Type		Description
功能	注册主题变更通知回调		
参数	callback	Callback	异步回调, 变更后的主题通过此回调进行通知

4.14.4.3 *Callback .onThemeChanged(int theme)*

Field	Object Type		Description
功能	通过此接口可以获知		
参数	theme	int	获取设置后的主题: 0-"静谧天成" 1- "静雅天成" 2- "默认主题"



4.15 V2I-LITE-FUN-REQ-4150001/A-getProvisioningState

4.15.1 Use cases

4.15.2 Requirements

4.15.2.1 V2I-LITE-REQ-4151521/ int getInstance

4.15.2.2 V2I-LITE-REQ-4151522/ int getProvisioningState()

4.15.3 White Box View

4.15.3.1 *Activity Diagrams*

4.15.3.2 *Sequence Diagrams*

4.15.4 Interfaces & Parameters

4.15.4.1 *getInstance(Context context)*

Field	Description
功能	获取 SystemInfoManager 的单例对象
参数	Context 上下文关系
返回	SystemInfoManager 单例对象

4.15.4.2 *int getProvisioningState()*

Field	Object Type	Description
功能	通过此接口可以获取当前 IVI 的 provision 状态	
参数	-	-
返回 值	-	int 状态至少包含 1: 成功 0:失败



4.16 V2I-LITE-FUN-REQ-4170001/A-APK installation & Configuration

4.16.1 Use cases

4.16.2 Requirements


4.16.2.1 V2I-LITE-REQ-4171721/ APK installation & Configuration

4.16.3 White Box View

4.16.3.1 Activity Diagrams

4.16.3.2 Sequence Diagrams

4.16.3.3 APK 安装与配置

Field	Description
功能	<p>功能一：添加测试工具的安装权限； 功能二：保证测试工具可以上网； 功能三：可以在原生 setting 中选择“模拟位置”；例如，在 phase 4 上可以模拟位置信息应用置为外装的 APK——TLSAPP。（（非 YFV 职责））</p> 

4.17 V2I-LITE-FUN-REQ-4180001/A-TEE

Trusted Execution Environment(TEE), applies to phase5.

**4.17.1 Use cases****4.17.2 Requirements****4.17.2.1** V2I-LITE-REQ-4181821/ TEE**4.17.3 White Box View****4.17.3.1** *Activity Diagrams***4.17.3.2** *Sequence Diagrams***4.17.4 Interfaces & Parameters****4.17.4.1** *TEE*

Alias Name	Key Name	作用	大小	存储方式	提供方
V2ICommunicationKeyAlias	V2ICommunicationKey	AES Key, “V2I APP”与“V2I Cloud”通信过程中的通信数据加解密用的 KEY。	32byte	由 V2I APP 调用 FCS 提供的接口存入。	由 “V2I APP”与“V2I Cloud”首次通信时"V2I Cloud"端生成。
V2ICityKeyAlias	V2ICityKey	AES Key, “V2I APP”与“V2I 城市数据提供商”通信过程中的通信数据加解密用的 KEY。	32byte	由 V2I APP 调用 FCS 提供的接口存入。	"V2I 城市数据提供商"生成后经过“V2I Cloud”下发给“V2I APP”。



5 Appendix: Reference Documents

Reference #	Document Title
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	