

特定 DTC 照会アプリ（Android 版）対応 VCI 規格

第1.4版

令和6年度

独立行政法人 自動車技術総合機構

変更履歴

版	内容	発行日
第 0.1 版	ドラフト版	令和 6 年 6 月 24 日
第 1 版	正式版	令和 6 年 7 月 31 日
第 1.1 版	自機工様、機工協様へのレビュー結果の反映 誤記訂正	令和 6 年 8 月 30 日
第 1.2 版	「5 Android 対応の VCI ドライバ規格の詳細 (3) プロトコルのサポート」にて、通信制御ミドルウェアが使用する API を判断するフローに間違いがあり修正 「5 Android 対応の VCI ドライバ規格の詳細 (6) 開発環境/リリース形態」にて、サポートする対応 OS の考え方を追加。また、C 言語で実装時、Google の公式見解に合わせたビルドの考え方を追記。	令和 6 年 9 月 30 日
第 1.3 版	「5 Android 対応の VCI ドライバ規格の詳細 (6) 開発環境/リリース形態」にて、aar ファイル名の規則を変更。 また、データアライメントの設定についての記載を追加。	令和 7 年 1 月 14 日
第 1.4 版	「5 Android 対応の VCI ドライバ規格の詳細 (6) 開発環境/リリース形態」にて、minSDK を 28 にすることを明記	令和 7 年 4 月 16 日

目次

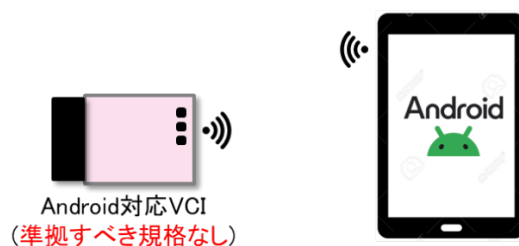
1	規格策定の背景	3
2	本書の目的	3
3	Android 端末と VCI との接続方法	4
4	Android 対応の VCI ドライバ規格の概要	5
5	Android 対応の VCI ドライバ規格の詳細	6
	(1) API 関数	6
	(2) 構造体変数名、定義値	8
	(3) プロトコルのサポート	9
	(4) レジストリの置き換え仕様	10
	(5) タイムスタンプ仕様	11
	(6) 開発環境/リリース形態	11
6	VCI ベンダの役割・責任分界点	14
	(1) Windows の場合	14
	(2) Android の場合	15
7	Android 対応 VCI ドライバの配布方法	16
8	Android のバージョンアップ時の対応	17
9	参考資料	19

1 規格策定の背景

現在、OBD 検査で使用される特定 DTC 照会アプリは、Windows PC と、SAE J2534 規格に準拠した VCI (Vehicle Communication Interface) でシステムが構成されています。



しかし昨今、Android などの情報端末の普及に伴い、これらの端末上で特定 DTC 照会アプリを動作させ、OBD 検査業務を行いたいというニーズが増えております。しかし、Android 端末と、VCI を接続するための規格が存在しておらず、対応する VCI が普及できない状況となっております。



2 本書の目的

自動車技術総合機構では、Android 端末を使った OBD 検査業務の実現を目指しております。そこで、OBD 検査に特化した、Android 端末と VCI の規格の策定を実施しています。

本書では、OBD 検査用 Android 対応 VCI の規格をまとめます。



3 Android 端末と VCI との接続方法

VCIと端末の間の具体的な接続方法は、Androidで、以下の通りとします。ただし、どの接続方式をVCIがサポートするかは、VCIベンダが決定するものとします。また、対応する接続方法に合わせて、VCIドライバとVCI間の接続仕様、実装仕様は、VCIベンダが決定するものとします。

○: 対応可能
△: 対応可能(但しハードルは高い)

接続方法	USB接続	Bluetooth接続	Wi-Fi接続
イメージ図 (Wi-Fiルータがない場合)			
イメージ図 (Wi-Fiルータがある場合)			
Android端末	○	○	○

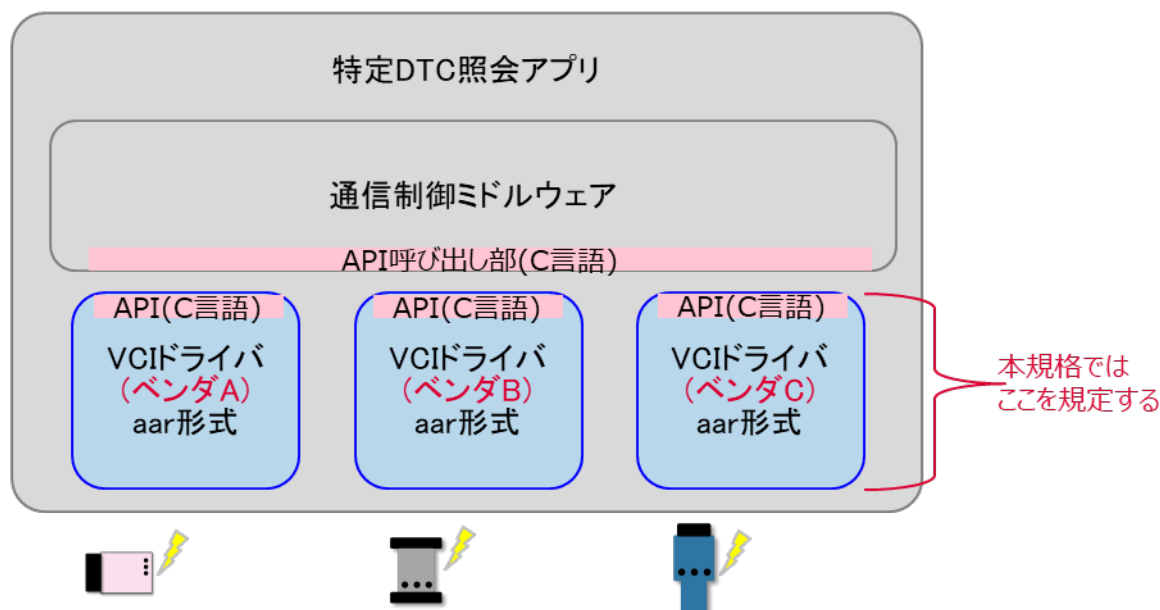
なおBluetooth接続の場合は、Classic接続か、BLE接続か、どこまでサポートするかは、VCIベンダが決定するものとします。

4 Android 対応の VCI ドライバ規格の概要

Windows用SAE J2534VCIドライバ規格に対して、OBD検査用Android対応VCIドライバ規格は以下の表に示す通りとなっています。

No.	区分	Windows	Android
1	用途	OBD 検査用、他 (Pass-Thru Vehicle Programming Tools / Vehicle Diagnostic Tools)	OBD 検査用
2	発行	SAE	自動車技術総合機構
3	規格書	J2534-1 Ver. 04. 04 J2534-2 Ver. 04. 04 J2534-1 Ver. 05. 00 J2534-2 Ver. 05. 00	左記規格の中で、OBD 検査用途として最低限必要な API、プロトコルに絞り込み、1 つの規格として本書でまとめる。
4	API 言語	Win32 API (C 言語)	API (C 言語)
5	API 関数	PassThru API	基本的に PassThru API Ver. 04. 04/Ver. 05. 00 の仕様に準じる。 但し” <u>PassThru</u> ・・・” で表現される関数名を OBD 検査用” <u>ePTI0404</u> ・・・”、” <u>ePTI0500</u> ・・・” に置き換える
6	ドライバのモジュール形式	動的ライブラリ (DLL)	静的ライブラリ (aar 形式 : <u>A</u> ndroid <u>A</u> rchive Library)
7	レジストリ	あり	なし(代替案で解決)
8	デバイスの検索	レジストリを参照	API を使い実施 (ePTIScanForDevices, ePTIGetNextDevice)
9	プロトコルサポート	レジストリを参照	API を使い取得
10	デバイス、メーカー名	レジストリを参照	API を使い取得

Androidの場合



5 Android 対応の VCI ドライバ規格の詳細

(1) API 関数

基本的には SAE J2534 の Ver.04.04 と Ver.05.00 の PassThru API 仕様に準じるものとします。ただし、“PassThru・・・”で表現される関数名を、OBD 検査用では“ePTI0404・・・”、“ePTI0500・・・”に置き換えるものとします。しかし、VCI の検索と指定に必要な PassThruScanForDevices および PassThruGetNextDevice は、Ver.04.04、Ver.05.00 で共通の関数であるため、API 名に 0404 や 0500 を付加しないものとします。

No.	準拠する J2534 API	準拠する J2534 仕様	OBD 検査用 API	備考
1	PassThruScanForDevices	Ver. 04. 04 Ver. 05. 00	ePTIScanForDevices	VCI ドライバでの実装およびサポートは必須とします
2	PassThruGetNextDevice	Ver. 04. 04 Ver. 05. 00	ePTIGetNextDevice	
3	PassThruOpen	Ver. 04. 04	ePTI0404open	
4	PassThruClose	Ver. 04. 04	ePTI0404Close	
5	PassThruConnect	Ver. 04. 04	ePTI0404Connect	
6	PassThruDisconnect	Ver. 04. 04	ePTI0404Disconnect	
7	PassThruReadMsgs	Ver. 04. 04	ePTI0404ReadMsgs	

8	PassThruWriteMsgs	Ver. 04. 04	ePTI0404WriteMsgs	
9	PassThruStartMsgFilter	Ver. 04. 04	ePTI0404StartMsgFilter	
10	PassThruStopMsgFilter	Ver. 04. 04	ePTI0404StopMsgFilter	
11	PassThruReadVersion	Ver. 04. 04	ePTI0404ReadVersion	
12	PassThruGetLastError	Ver. 04. 04	ePTI0404GetLastError	
13	PassThruIoctl	Ver. 04. 04	ePTI0404Ioctl	
14	PassThruOpen	Ver. 05. 00	ePTI05000pen	
15	PassThruClose	Ver. 05. 00	ePTI0500Close	VCI ドライバ での実装は必 須とします が、サポート は任意としま す。 サポートしてい ない場合は未サ ポートエラー (NOT SUPPORTED) を返 すこととします
16	PassThruConnect	Ver. 05. 00	ePTI0500Connect	
17	PassThruDisconnect	Ver. 05. 00	ePTI0500Disconnect	
18	PassThruLogicalConnect	Ver. 05. 00	ePTI0500LogicalConnect	
19	PassThruLogicalDisconnect	Ver. 05. 00	ePTI0500LogicalDisconnect	
20	PassThruSelect	Ver. 05. 00	ePTI0500Select	
21	PassThruReadMsgs	Ver. 05. 00	ePTI0500ReadMsgs	
22	PassThruQueueMsgs	Ver. 05. 00	ePTI0500QueueMsgs	
23	PassThruStartMsgFilter	Ver. 05. 00	ePTI0500StartMsgFilter	
24	PassThruStopMsgFilter	Ver. 05. 00	ePTI0500StopMsgFilter	
25	PassThruReadVersion	Ver. 05. 00	ePTI0500ReadVersion	
26	PassThruGetLastError	Ver. 05. 00	ePTI0500GetLastError	
27	PassThruIoctl	Ver. 05. 00	ePTI0500Ioctl	

各 API 関数の引数および仕様は、準拠している SAE J2534 の API 関数に準拠するものとします。以下が特定 DTC 照会アプリで参照している SAE J2534 仕様となります。

<Ver.04.04>

SAE J2534-1 (Revised: DEC2004)

SAE J2534-2 (Revised: DEC2020)

<Ver.05.00>

SAE J2534-1_0500 (Issued: JAN2022)

SAE J2534-2_0500 (Issued: JAN2022)

なお ePTIGetNextDevice で得られる SDEVICE の DeviceName は、以下のフォーマットとします。文字数は 80 文字以下(終端文字 NULL を含む)とします。

[製品名]_[製造番号]_[接続方式]

(例) Acme_500_USB

ここで、接続方式のフォーマットは以下の通りとします。

USB の場合 : USB

Bluetooth の場合 : BT

Wi-Fi の場合 : Wi-Fi

なお、各 API でサポートする引数およびプロトコルは、自動車技術総合機構発行の「特定 DTC 照会アプリインターフェース仕様書」(OBD ポータルで公開)に準拠するものとします。

(2) 構造体変数名、定義値

基本的には SAE J2534 の Ver.04.04 と Ver.05.00 の PassThru API の構造体変数名に準じるものとします。ただし、Ver.04.04 と Ver.05.00 で同じ構造体変数名であるが、メンバー構成が異なる場合は、変数名(具体的には PASSTHRU_MSG)には”0404”および”0500”を付加して区別するものとします。

No.	J2534 変数名	OBD 検査用変数名		備考
		Ver. 04.04	Ver. 05.00	
1	SDEVICE	SDEVICE		ePTIScanForDevices/ ePTIGetNextDeviceで使用
2	PASSTHRU_MSG	PASSTHRU0404_MSG	PASSTHRU0500_MSG	メッセージ送受信、フィルタ設定で使用
3	SCONFIG_LIST	SCONFIG_LIST		SPARAM_LIST と同じ、 ePTI0404Ioctl および ePTI0500Ioctl で使用
4	SCONFIG	SCONFIG		SPARAM と同じ、 ePTI0404Ioctl および ePTI0500Ioctl で使用
5	SBYTE_ARRAY	SBYTE_ARRAY		ePTI0404Ioctl および ePTI0500Ioctl で使用
6	RESOURCE_STRUCTURE	-	RESOURCE_STRUCT	ePTI0500Connect および ePTI0500Ioctl で使用
7	ISO15765_CHANNEL_DESCRIPTOR	-	ISO15765_CHANNEL_DESCRIPTOR	ePTI0500LogicalConnect で使用

8	SCHANNELSET	-	SCHANNELSET	ePTI0500Select で使用
9	IP4_CHANNEL_DESCRIPTOR	-	IP4_CHANNEL_DESCRIPTOR OR	ePTI0500LogicalConnect で使用
10	UDP_CHANNEL_DESCRIPTOR	-	UDP_CHANNEL_DESCRIPTOR OR	ePTI0500LogicalConnect で使用
11	TCP_CLIENT_CHANNEL_DESCRIPTOR	-	TCP_CLIENT_CHANNEL_DESCRIPTOR	ePTI0500LogicalConnect で使用
12	SPARAM_LIST	SPARAM_LIST		ePTI0404Ioctl (GET_DEVICE_INFO) および ePTI0500Ioctl (GET_DEVICE_INFO) で使用
13	SPARAM	SPARAM		ePTI0404Ioctl (GET_DEVICE_INFO) および ePTI0500Ioctl (GET_DEVICE_INFO) で使用

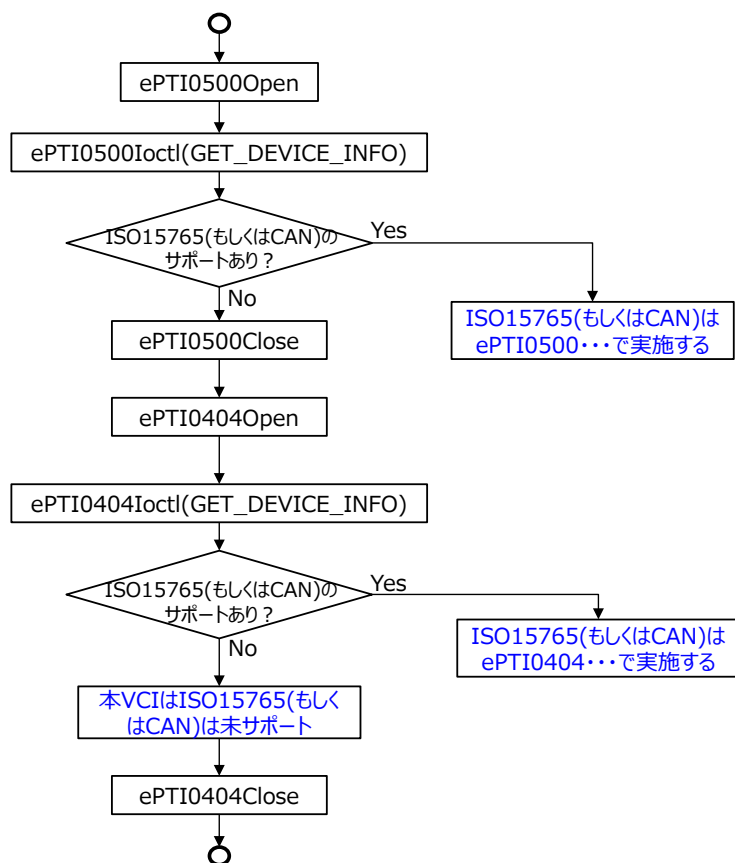
また、J2534 の各種定義値(エラーコード、プロトコル指定など)は、Android 対応 VCI ドライバ規格では、そのまま採用するものとします。

(3) プロトコルのサポート

プロトコル毎に使用する OBD 検査用 API は以下の通りとします。

No.	プロトコル名	特定 DTC 照会アプリがサポートする J2534 仕様	OBD 検査用 API	備考
1	ISO9141	Ver. 04.04	ePTI0404 . . .	サポートは任意とします
2	ISO14230	Ver. 04.04	ePTI0404 . . .	
3	CAN	Ver. 04.04 Ver. 05.00	ePTI0404 . . . ePTI0500 . . .	通信制御ミドルウェアがどちらの API を使用するかは、以下のフローで判断
4	ISO15765	Ver. 04.04 Ver. 05.00	ePTI0404 . . . ePTI0500 . . .	
5	Ethernet	Ver. 05.00	ePTI0500 . . .	サポートは任意とします

また、CAN および ISO15765 の通信を、Ver.04.04 もしくは Ver.05.00 のどちらの API で実施するか判断するため、通信制御ミドルウェアは以下のフローで判断するものとします。



また、Ethernet プロトコルのサポートも、ePTI0500Ioctl(GET_DEVICE_INFO)のサポート取得で通信制御ミドルウェアは判断するものとします。

(4) レジストリの置き換え仕様

Android には Windows にあるレジストリが存在しないため、その代替え策を以下の通り規定します。

No.	区分	Windows	Android
1	ベンター名	レジストリ の” Vender” より取得	aar ファイル名から取得
2	デバイス名	レジストリの” Name” より取得	aar ファイル名から取得
3	ドライバーパス	レ ジ ス ト リ の” FunctionLibrar y” より取得	静的にリンクするのでパス 情報は不要
4	シリアル No 取	デ フ ォ ル ト は	0x30000

	得 ID	0x30000。変更の場合はレジストリの”GET_VCI_SERIAL_NO”より取得	
5	型式取得 ID	デフォルトは 0x30001。変更の場合はレジストリの”GET_VCI_TYPE”より取得	0x30001

(5) タイムスタンプ仕様

VCI 内では、32bit のフリーランニングカウンタ(Lsb=1 μ s、1ms)を存在しており、これを使用して経過時間を計算しています。通常では

$$\text{経過時間} = \text{最新取得タイムスタンプ} - \text{基準の取得タイムスタンプ}$$

で計算しています。このタイムスタンプは、Windows の Win32 アプリケーションでは、unsigned long 型の変数に格納されていることがあります。

しかし、Win32 アプリケーションと Android のアプリでは、以下の通り unsigned long 型のビット数に違いがあります。

型	Windows (Win32)	Android
unsigned long	32bit	64bit

従い、Windows で使用している C 言語の J2534 ドライバのソースコードを流用して Android ドライバを開発する場合、上記の型の違いに注意しないと、正しい経過時間が計算されない可能性があります。つまり、Android の 64bit の unsigned long 型の上位 32bit をゼロ埋めし、32bit の符号なし変数へ変換して、経過時間を計算する必要があります。

(6) 開発環境/リリース形態

開発環境および VCI ドライバのリリース形態は、以下の通りとします。

No.	区分	Android
1	対応 OS	Android 13、Android 14、Android 15 (※1)
2	開発環境	Android Studio とします。Android Studio で使用する SDK バージョンは最新の SDK34 となります。 ただし SDK は毎年バージョンアップするため、VCI ドライバを SDK バージョンアッ

		ブ毎にビルドし直す必要があるかどうかは、開発言語で異なります。 ◆開発言語が C 言語のみ使用している場合 VCI ドライバに変更がない場合、SDK のバージョンアップに伴いビルドし直す必要はありませんが、既にリリースしているドライバの SDK バージョンが Google の保証するバージョンから外れた場合は、新しい SDK でビルドし直し、リリースする必要があります。(※2) ◆開発言語が C 言語以外 (Java, Kotlin) を使用している場合 VCI ドライバに変更がない場合でも、SDK のバージョンアップに伴いビルドし直す必要があります。 ◆minSDK 開発言語に関わらず、minSDK は 28 にすることとする。
3	ビルド時のオプション等の規定	VCI ドライバは、開発言語によって、難読化対応の必要性有無や、使用して良いビルドオプションが変更となります。 ◆開発言語が C 言語のみ使用している場合 ◇使用可能ビルドオプション -s -O3 -fomit-frame-pointer -finline-functions ◇使用不可ビルドオプション -fvisibility=hidden ◆開発言語が C 言語以外 (Java, Kotlin) を使用している場合 ◇設定ファイル「build.gradle.kts」内

		の設定値について “isMinifyEnabled” および” isShrinkResources” は False とする ◇設定ファイル「proguard-android- optimize.txt」および「proguard- rules.pro」について Android Studio プロジェクト立ち上げ時 に生成されるデフォルト状態のまま変更 しないこと(全行コメントアウトの状態)
4	リリース形態	各 VCI ベンダは VCI ドライバを aar ファ イルとし、窓口機関に提出する。aar ファ イル名は既定に沿った命名とします。 ◆窓口機関： 自動車技術総合機構 ◆ファイル名：以下の通りとします vcidriver_<ベンダ名>_<デバイス名 >.aar ※so ファイルのファイル名設定は変更 しないこと。 ◆受け入れられる VCI ドライバ VCI ベンダが開発し窓口機関に提出した ものに限るものとします
5	その他	データアライメントの設定はデフォルト のまま変更しないこと。

(※1) サポートする Android のバージョンは、新しい Android バージョンのリリースに合わ
せて毎年見直すものとします。また古い Android バージョンについては、Google の
サポートが終わり次第、VCI ドライバのサポート範囲からも除外するものとします。
見直した Android の対応バージョンについては、窓口機関である自動車技術総合機
構から VCI ベンダに通知するものとします。VCI ベンダは、「8 Android のバージ
ョンアップ時の対応」に示す対応を実施するものとします。

(※2) Google の公式見解では、最新リリースの 1 つ前までの API(SDK)しかサポートしない
となっています。従い古い SDK でビルドされたドライバは、Google Play の審査が通
らない可能性があります。毎年の Android の新しいバージョンのリリースに合わせて、
Google の保証する API(SDK)を以下の URL で確認の上、保証範囲外の SDK でビル

ドされている VCI ドライバは、たとえ VCI ドライバに変更がない場合でも、新しい SDK でビルドし直しリリースするものとします。この場合、VCI ドライバのバージョンは必ずアップデートするものとし、区別が出来るものとします。

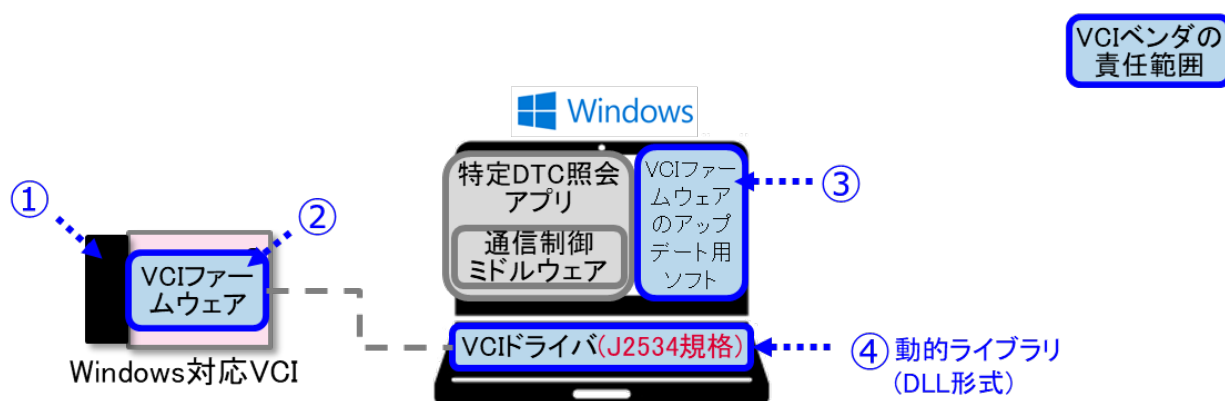
Google 公式見解 URL

<https://support.google.com/googleplay/android-developer/answer/11926878?hl=ja>

6 VCI ベンダの役割・責任分界点

(1) Windows の場合

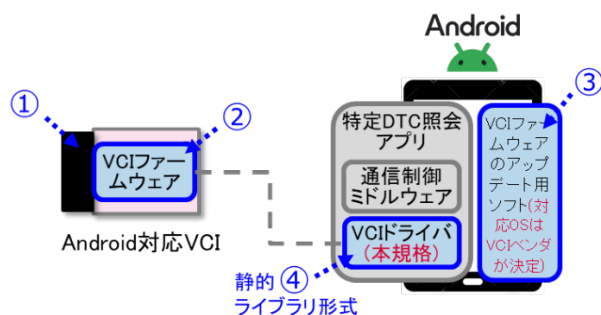
Windows 対応 VCI を製造・販売する VCI ベンダは、以下を提供するものとします。



#	名称	内容	備考
①	VCIハードウェア	VCIハードウェア本体や接続用ケーブルなどの周辺機器	
②	VCIファームウェア	VCIの中で動作し、VCIの制御を司る組み込みソフトウェア	
③	VCIファームウェアのアップデート用ソフトウェア	VCIのファームウェアをアップデートする時に使用するソフトウェア	VCIベンダがエンドユーザに直接配布し、エンドユーザ自身でファームウェアをアップデートする
④	VCIドライバ	VCIを使用してダイアグ通信の送受信を行うために必要なドライバ。SAE J2534 APIに準拠している	動的ライブラリ形式のため、特定DTC照会アプリとは独立して、VCIベンダ自身がエンドユーザに直接配布する

(2) Android の場合

Android 対応 VCI を製造・販売する VCI ベンダも、Windows と同様に、以下を提供するものとします。

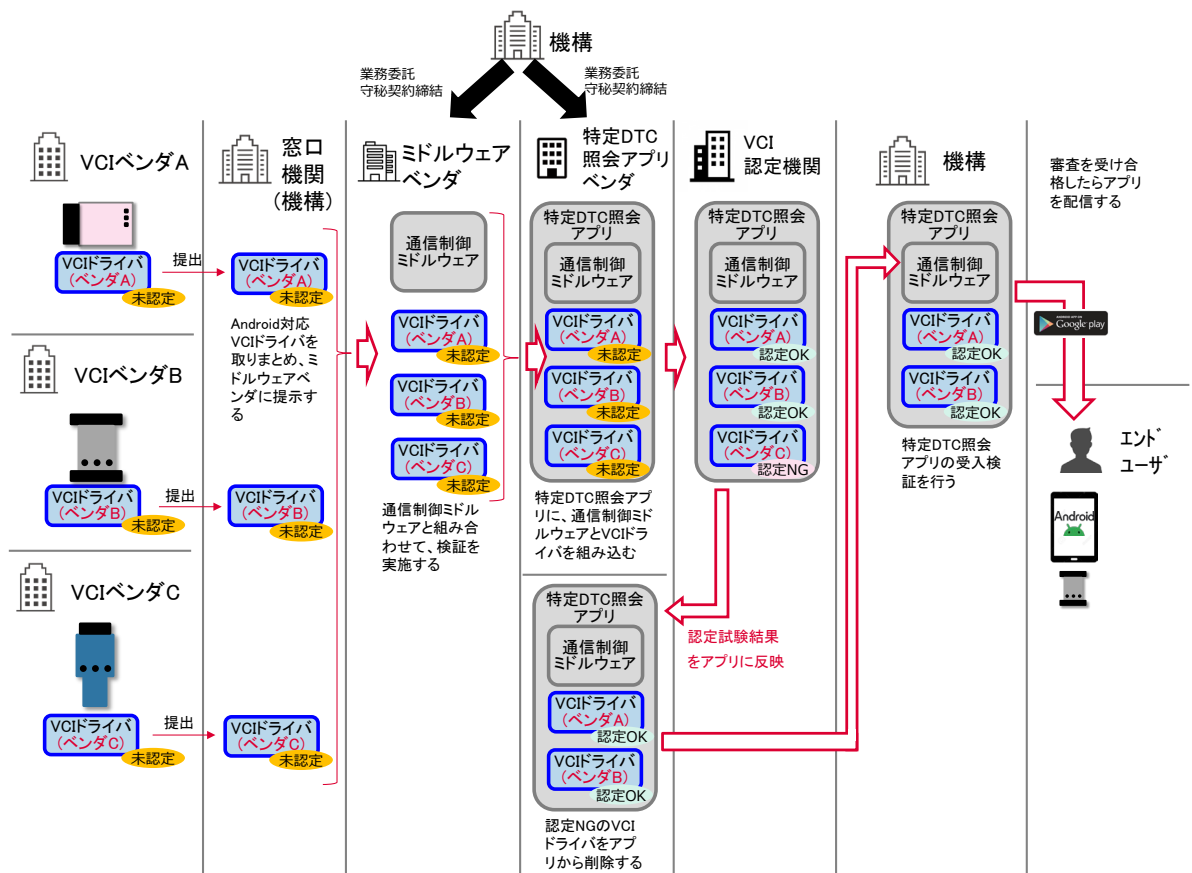


#	名称	内容	備考
①	VCIハードウェア	VCIハードウェア本体や接続用ケーブルなどの周辺機器	
②	VCIファームウェア	VCIの中で動作し、VCIの制御を司る組み込みソフトウェア	
③	VCIファームウェアのアップデート用ソフトウェア	VCIのファームウェアをアップデートする時に使用するソフトウェア	VCIベンダがエンドユーザに直接配布し、エンドユーザ自身でファームウェアをアップデートする。ただし本ソフトの対応OSはVCIベンダが決めるものとします(例:Windows版のみの提供でも可とします)
④	VCIドライバ	VCIを使用してダイアグ通信の送受信を行うために必要なドライバ。今回規定する規格に準拠している	静的ライブラリ形式のため、VCIベンダ自身がエンドユーザに直接配布できず、特定DTC照会アプリに同梱して提供する必要がある

7 Android 対応 VCI ドライバの配布方法

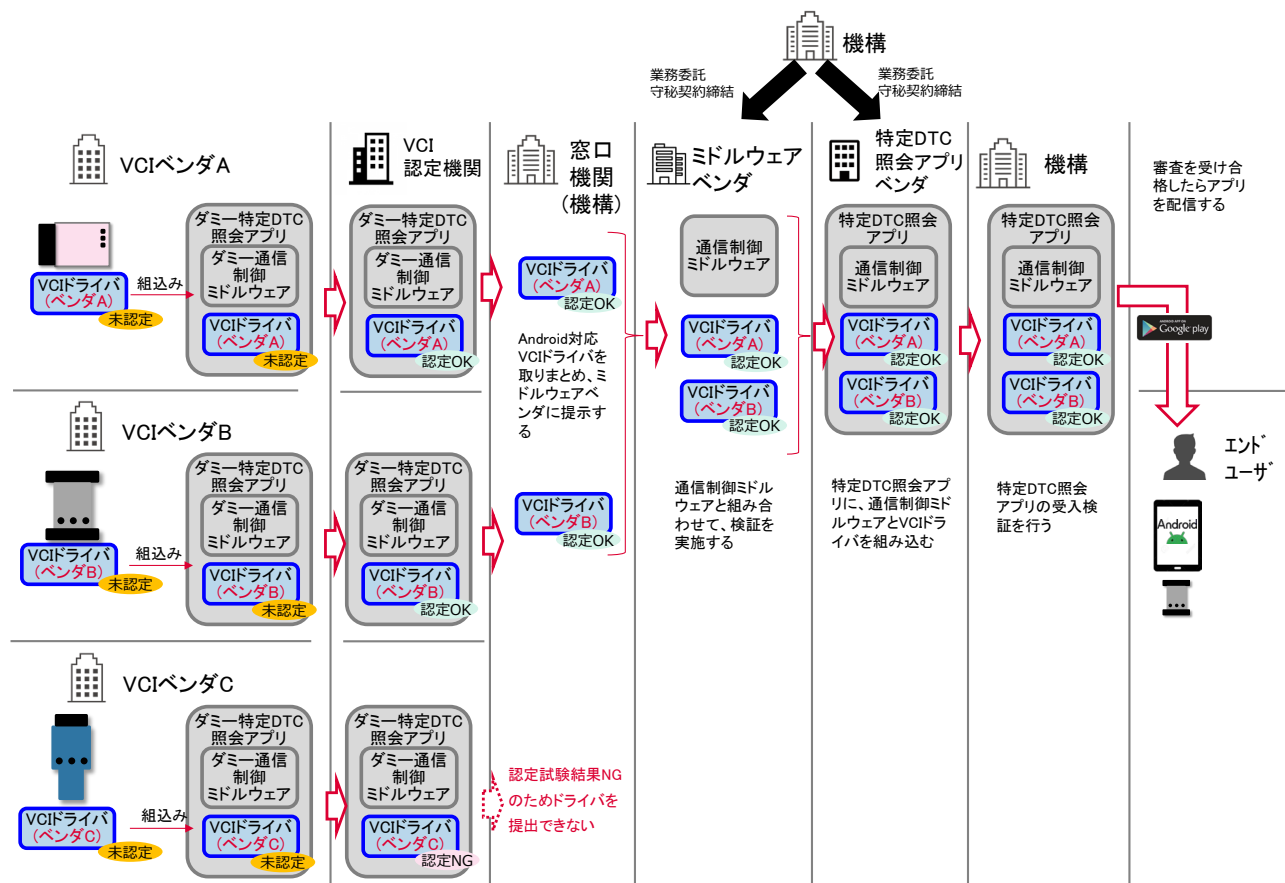
Android 対応 VCI ドライバは、以下のルートで特定 DTC 照会アプリに組み込まれ、エンドユーザに配布されるものとします。このため、VCI ドライバの不具合で修正版 VCI ドライバの配布が必要となった場合、特定 DTC 照会アプリのバージョンアップ計画に合わせてのみ展開されることを基本ルールとします。(ユーザへの影響が大きい不具合の場合は、別途調整するものとします。)

<特定 DTC 照会アプリ 立ち上げ時のフロー>



なお、特定 DTC 照会アプリの立ち上げが完了し、定常開発となった後は、VCI ベンダがダミーの特定 DTC 照会アプリを使い認定試験を行える運用に切り替えていくことを検討していくものとします。

<特定 DTC 照会アプリの定常開発時のフロー>



8 Android のバージョンアップ時の対応

Android 対応 VCI ドライバについて、OS のバージョンアップ時は、以下の基本ルールに従い、VCI ベンダは評価および品質保証を行うこととします。

OS	メジャーバージョンアップ時	マイナーバージョンアップ時
Windows	バージョンアップ頻度は数年に1回程度 (例: Windows 10→Windows 11) この時のVCIドライバの評価は必須とします また、事前にMicrosoftから提供される 新バージョンの評価版Windowsで検証する ものとします	Windows Updateによるバージョンア ップ頻度は随時 この時のVCIドライバの評価は任意と します
Android	バージョンアップ頻度は1年に1回程度	セキュリティ対応のバージョンアップ

OS	メジャーバージョンアップ時	マイナーバージョンアップ時
	この時のVCIドライバの評価は必須とします なお、Androidは端末メーカー各社がカスタマイズしており、全てのカスタマイズバージョンまで検証するのは不可能であるため、事前にGoogleから提供される新バージョンの評価版Androidで検証するものとします	頻度は随時 この時のVCIドライバの評価は任意とします

なお、VCI ドライバの最新バージョンの OS への追従は、VCI ベンダの判断にゆだねるものとしませんが、追従をしない場合は、窓口機関に報告するものとします。窓口機関から報告を受けた通信制御ミドルウェアベンダもしくは特定 DTC 照会アプリベンダは、追従をしない VCI を特定 DTC 照会アプリから選択できないよう、VCI ドライバを削除するものとします。

9 参考資料

本仕様に関係する資料は以下の通りである。

SAE J2534-1 (Revised: DEC2004)	SAE 発行
SAE J2534-2 (Revised: DEC2020)	SAE 発行
SAE J2534-1_0500 (Issued: JAN2022)	SAE 発行
SAE J2534-2_0500 (Issued: JAN2022)	SAE 発行
特定 DTC 照会アプリインターフェース仕様書	自動車技術総合機構発行