特定 DTC 照会アプリ（Android 版）対応 VCI 規格

第1.9版

令和８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 1 版	正式版	令和 6 年 7 月 31 日
第 1.2 版	「５Android 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	令和 6 年 9 月 30 日
	「５Android 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。	
第 1.3 版	「５Android 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	「５Android 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
第 1.5 版	「４Android 対応の VCI ドライバ規格の概要」の図に補足説明追加	令和 7 年 5 月 9 日
第 1.6 版	「５Android 対応の VCI ドライバ規格の詳細 (4)レジストリの置き換え仕様」にて、No.1　ベンダ名と No.2　デバイス名の Android の代替え策の記載を変更	令和 7 年 6 月 20 日
	「５Android 対応の VCI ドライバ規格の詳細 (6)開発環境/リリース形態 No.4 リリース形態」にて、“※so ファイルのファイル名設定は変更しないこと。”を削除	
第 1.7 版	「４Android 対応の VCI ドライバ規格の概要　Windows と Android 相違表」にて、”No.11　Android 環境情報”項目を追加	令和 7 年 7 月 25 日
	「５Android 対応の VCI ドライバ規格の詳細 (1)API 関数」API 関数表に No.28,29,30 の API 追加	
	「５Android 対応の VCI ドライバ規格の詳細　(6)Android 環境情報取得仕様」を追加	
	「５Android 対応の VCI ドライバ規格の詳細 (7)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表 No.1 対応 OS の記載を修正	
第 1.8 版	文書全体に記載している表記を以下の通り修正 「VCI ベンダ」から「スキャンツールメーカー」に修正	令和 7 年 11 月 7 日

	「５Android 対応の VCI ドライバ規格の詳細(7)開発環境/リリース形態」にて、NDK バージョンに関する記載を追加	
	「５Android 対応の VCI ドライバ規格の詳細 (8)マニフェストガイドライン、(9)リソース定義ガイドライン、(10)OSS(オープンソースソフトウェア)利用ガイドライン」を追加	
	「８Android のバージョンアップ時の対応」にて、VCI ドライバが最新バージョン OS に追従しない場合の対応として、VCI ドライバを削除できる事業者の記載から「ミドルウェア開発・保守事業者」を削除	
第 1.9 版	「３Android 端末と VCI との接続方法」に「(2)Android 端末と VCI デバイスを接続する場合に必要な定義」を追加	令和 8 年 1 月 16 日
	「５Android 対応の VCI ドライバ規格の詳細 (4)レジストリの置き換え仕様」の so ファイル名からのベンダ名・デバイス名取得する方法を削除し、ソースコードへの定義を必須化	
	「５Android 対応の VCI ドライバ規格の詳細 (11)グローバル領域記載定義」を追加	
	「１０商標について」を追加	
	その他誤記訂正	

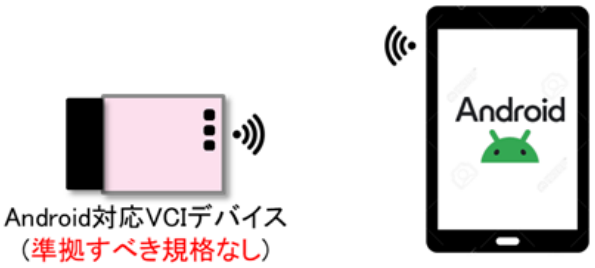
目次

１　規格策定の背景 4
２　本書の目的 4
３　Android 端末と VCI デバイスとの接続方法 5
(1)対応可能な接続方法 5
(2)Android 端末と VCI デバイスを接続する場合に必要な定義と留意点 5
４　Android 対応の VCI ドライバ規格の概要 6
５　Android 対応の VCI ドライバ規格の詳細 7
(1)API 関数 7
(2)構造体変数名、定義値 9
(3)プロトコルのサポート 10
(4)レジストリの置き換え仕様 11
(5)タイムスタンプ仕様 12
(6)Android 環境情報取得仕様 12
(7)開発環境/リリース形態 15
(8)マニフェストガイドライン 17
(9)リソース定義ガイドライン 17
(10)OSS(オープンソースソフトウェア)利用ガイドライン 17
(11)グローバル領域記載定義 18
６　スキャンツールメーカーの役割・責任分界点 22
(1)Windows の場合 22
(2)Android の場合 23
７　Android 対応 VCI ドライバの配布方法 24
８　Android のバージョンアップ時の対応 25
９　参考資料 27
１０商標について 27

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る。

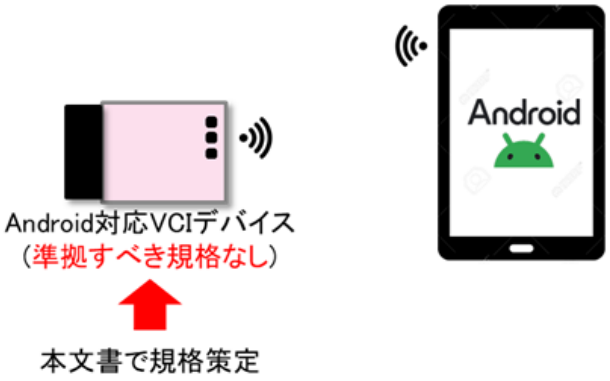
しかし昨今、Android などの情報端末の普及に伴い、これらの端末上で特定 DTC 照会アプリを動作させ、OBD 検査業務を行いたいというニーズが増加している。しかし、Android 端末と、VCI デバイスを接続するための規格が存在しておらず、対応する VCI デバイスが普及できない状況にある。



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的とする。そこで、OBD 検査に特化した、Android 端末と VCI デバイスの規格の策定を実施する。

本書では、 OBD 検査用 Android 対応 VCI デバイスの規格をまとめる。

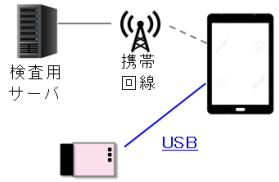
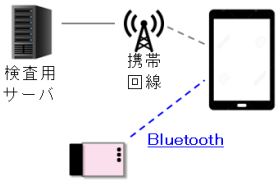
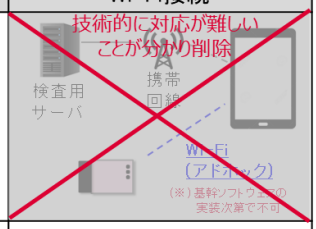
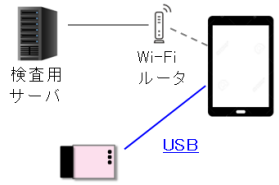
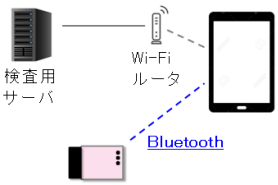
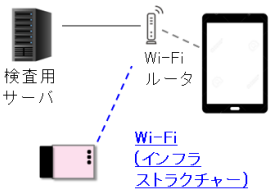


3　Android 端末と VCI デバイスとの接続方法

(1)対応可能な接続方法

VCIデバイスと端末の間の具体的な接続方法は、Androidで、以下の通りとする。ただし、どの接続方式をVCIデバイスがサポートするかは、スキャンツールメーカーが決定するものとする。また、対応する接続方法に合わせて、VCIドライバとVCIデバイス間の接続仕様、実装仕様は、スキャンツールメーカーが決定するものとする。

○: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、スキャンツールメーカーが決定するものとする。

(2)Android 端末と VCI デバイスを接続する場合に必要な定義と留意点

1.Bluetooth接続の留意点

VCI ドライバは、Bluetooth 接続の権限が取得できていることを確認してから接続する必要があるため、権限が取得できていない状態では Bluetooth 接続を行わないものとする。

2.USB接続に必要な定義

Android端末にVCIデバイスをUSB で接続するには、USB 接続したVCIデバイスへのアクセス権限を使用者に要求する必要があり、USB接続されているVCIデバイスを特定するための情報としてVCIデバイスのベンダID (idVendor)およびプロダクトID(idProduct)をグローバル領域に記載する。

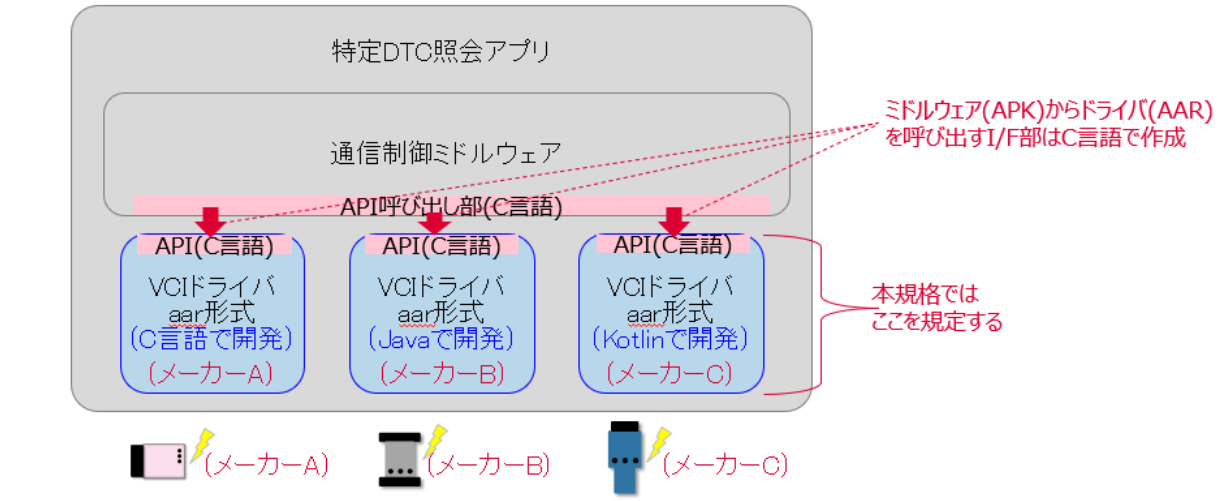
記載方法は、「５(11)グローバル領域記載定義」を参照。

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。

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”PassThru・・・”で表現される関数名を OBD 検査用”ePTI0404・・・”、”ePTI0500・・・”に置き換える。
6	VCI 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：Android Archive Library)
7	レジストリ	あり	なし(代替案で解決)
8	VCI 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	VCI デバイス、メーカー名	レジストリを参照	API を使い取得
11	Android 環境情報	対象外	VCI ドライバは API を介してミドルウェアから Android 環境情報(Context, JavaVM, JNIEnv)を取得できる。 (ePTISetAndroidContext, ePTISetAndroidJavaVM, ePTISetAndroidJNIEnv)

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する。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する。なお、VCI デバイス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する。

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する。
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	ePTI0500Open	
15	PassThruClose	Ver.05.00	ePTI0500Close	
16	PassThruConnect	Ver.05.00	ePTI0500Connect	VCI ドライバでの実装は必須とするが、サポートは任意とする。 サポートしていない場合は未サポートエラー(NOT SUPPORTED)を返すこととする。
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	
28	-	-	ePTISetAndroidContext	実装は任意とする。 VCI ドライバで Android 環境情報(Context, JavaVM, JNIEnv の値)を取得する場合は実装する。 詳細は「(6)Android 環境情報取得仕様」参照。
29	-	-	ePTISetAndroidJavaVM	
30	-	-	ePTISetAndroidJNIEnv	

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する。以下に特定 DTC 照会アプリで参照している SAE J2534 仕様を示す。

<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する。文字数は 80 文字以下(終端文字 NULL を含む)とする。

[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する。

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)に準拠するものとする。

(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する。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する。

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_STRUCT	-	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	ePTI0500LogicalConnect で使用
10	UDP_CHANNEL_DESCRIPTOR	-	UDP_CHANNEL_DESCRIPTOR	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する。

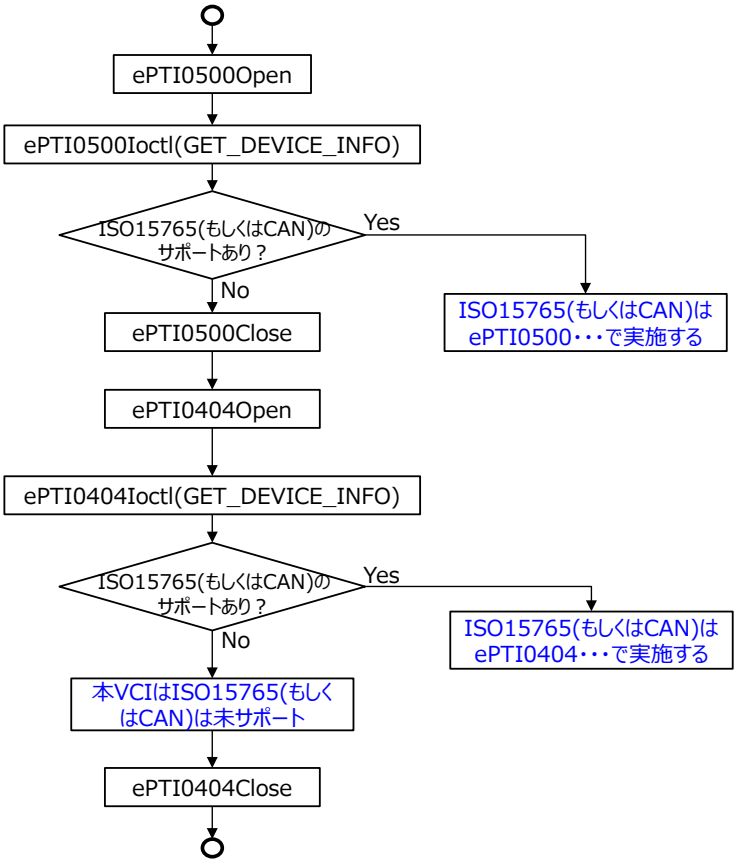
(3) プロトコルのサポート

プロトコル毎に使用する OBD 検査用 API は以下の通りとする。

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・・・	サポートは任意とする。
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
4	ISO15765	Ver.04.04	ePTI0404・・・	

		Ver. 05. 00	ePTI0500・・・	るかは、以下のフローで判断。
5	Ethernet	Ver. 05. 00	ePTI0500・・・	サポートは任意とする。

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する。



また、Ethernet プロトコルのサポートも、ePTI0500Ioctrl(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する。

(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する。

No.	区分	Windows	Android
1	メーカー名	レジストリの”Vendor”より取得	ソースコードに定義された情報から取得。ソースコード記載方法は、「(11)グローバル領域記載定義」を

			参照。
2	デバイス名	レジストリの”Name”より取得	ソースコードに定義された情報から取得。ソースコード記載方法は、「(11)グローバル領域記載定義」を参照。
3	ドライバーパス	レジストリの”FunctionLibrary”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 取得 ID	デフォルトは 0x30000。変更の場合はレジストリの”GET_VCI_SERIAL_NO”より取得。	0x30000
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 デバイス内では、32bit のフリーランニングカウンタ(Lsb=1μ、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る。通常では

経過時間　＝　最新取得タイムスタンプ　-　基準の取得タイムスタンプ

で計算している。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る。

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る。

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る。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る。

(6)Android 環境情報取得仕様

Android のシステム機能・リソース・サービスを利用する場合、Context, JavaVM, JNIEnv の値が必要となることがある。VCI ドライバで利用する場合、VCI ドライバ側で下記 API を実装することで、ミドルウェアから値を受け取ることが可能。

なお、VCI ドライバで Context, JavaVM, JNIEnv の値が不要な場合は、VCI ドライバへの本

API の実装は不要。

■API 仕様

(6).1 ePTISetAndroidContext

(6).1.1 API 概要

ePTISetAndroidContext の概要について、下図に示す。

関数名	ePTISetAndroidContext		
機能	VCI ドライバへシステム機能・リソース・サービスにアクセスするための「Context」の値が渡される。 VCI ドライバがロードされた時に Context の値が渡される。		
前提条件	なし。		
	Input/Output	型	説明
引数#1	<Input>	jobject	Android 各機能アクセス用アプリケーションコンテキスト
戻り値	<Output>	long	エラーコード

(6).1.2 引数補足説明

ePTISetAndroidContext の引数について以下に示す。

引数説明	VCI ドライバ内でシステム機能・リソース・サービスにアクセスするための Context の値。

(6).1.3 戻り値補足説明

ePTISetAndroidContext の戻り値について以下に示す。

戻り値説明	下表のエラーコードの値を返す。

分類	エラーコード(10 進数)
正常	0
未サポートエラー	1
その他エラー	7

(6).2 ePTISetAndroidJavaVM

(6).2.1 API 概要

ePTISetAndroidJavaVM の概要について以下に示す。

関数名	ePTISetAndroidJavaVM		
機能	C 言語から C 言語以外の関数を呼び出す場合に使用する「JavaVM」の値が渡される。 VCI ドライバがロードされた時に JavaVM の値が渡される。		
前提条件	なし。		
	Input/Output	型	説明
引数#1	<Input>	JavaVM*	Android Java 仮想マシンインターフェースへのポインタ
戻り値	<Output>	long	エラーコード

(6).2.2 引数補足説明

ePTISetAndroidJavaVM の引数について以下に示す。

引数説明	C 言語から C 言語以外の関数を呼び出すための Android Java 仮想マシンインターフェースポインタ。

(6).2.3 戻り値補足説明

ePTISetAndroidJavaVM の戻り値について以下に示す。

戻り値説明	下表のエラーコードの値を返す。

分類	エラーコード(10 進数)
正常	0
未サポートエラー	1
その他エラー	7

(6).3 ePTISetAndroidJNIEnv

(6).3.1 API 概要

ePTISetAndroidJNIEnv の概要について以下に示す。

関数名	ePTISetAndroidJNIEnv		
機能	C 言語から C 言語以外の関数を呼び出す場合に使用する「JNIEnv」の値が渡される。 VCI ドライバがロードされた時および、値の変更時に JNIEnv の値が渡される。		
前提条件	なし。		
	Input/Output	型	説明
引数#1	<Input>	JNIEnv*	Java Native インターフェースへのポインタ
戻り値	<Output>	long	エラーコード

(6).3.2 引数補足説明

ePTISetAndroidJNIEnv の引数について以下に示す。

引数説明	C 言語から C 言語以外の関数を呼び出すための Java Native インターフェースポインタ。

(6).3.3 戻り値補足説明

ePTISetAndroidJNIEnv の戻り値について以下に示す。

戻り値説明	下表のエラーコードの値を返す。

分類	エラーコード(10 進数)
正常	0
未サポートエラー	1
その他エラー	7

(7)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、以下の通りとする。

No.	区分		Android
1	対応 OS		Android 14、Android 15、Android 16（※1）
2	開発環境	開発に使うソフトウェア	Android Studio とする。
		SDK	Android Studio で使用する SDK バージョンは最新の SDK34 とする。 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 バージョンアップ毎にビルドし直す必要があるかどうかは、開発言語で異なる。 ◆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ないが、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る。(※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る。
		NDK	Google Play Store 要件を満たす正式リリースバージョンとする。
		minSDK	開発言語に関わらず、minSDK は 28 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る。 ◆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 VCI ドライバ開発用の AndroidStudio 新規プロジェクトを作成する際に自動的に生成されるものに対し、変更を加えない。（全行コメントアウトの状態）
4	リリース形態	各スキャンツールメーカーは 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する。 ◆窓口機関： 自動車技術総合機構 ◆ファイル名：以下の通りとする vcidriver_<メーカー名>_<デバイス名>.aar ◆受け入られる VCI ドライバ スキャンツールメーカーが開発し窓口機関に提出したものに限るものとする。
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。

(※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する。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する。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スキャンツールメーカーに通知するものとする。スキャンツールメーカーは、「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する。

(※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る。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る。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する。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する。

Google 公式見解 URL

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

(8) マニフェストガイドライン

マニフェストのガイドラインは、以下の通りとする。

項目	定義詳細
マニフェストの定義内容	・端末と VCI 接続に必要な定義のみ可能。 ・Wi-Fi、Bluetooth、USB の接続は定義済み。
マニフェストの定義場所	・特定 DTC 照会アプリでマニフェストを一元管理するため、VCI ドライバではマニフェスト定義はしない。 ・不足する定義は、スキャンツールメーカーより申請を行う。（申請方法は別途案内予定）

(9) リソース定義ガイドライン

VCI 通信に必要の無いリソース定義は行わない。

(10) OSS（オープンソースソフトウェア）利用ガイドライン

OSS を利用するためのガイドラインは、以下の通りとする。

項目	定義詳細
使用許可ライセンス	ソースコード開示が不要な BSD ライセンスのみ許可とする。
利用する OSS 情報の提供	スキャンツールメーカーは VCI ドライバで利用している OSS ライセンス情報を、OBD 検査ポータル（https://www.obd.naltec.go.jp/）の、「疎通確認・認定試験用 Android 版特定 DTC 照会アプリ作成依頼書」を取得して記載する。（※1） OSS を使用しているか確認する場合、Google 公式サービス等を利用し確認可能。 参考： Google 公式 OSS Licenses Plugin の導入と設定方法の URL https://developers.google.com/android/guides/opensource?utm_source=chatgpt.com&hl=ja

(※1)以下の項目に該当する場合は、OSS を使用している可能性が高いと考えられる。

・他社やネットから直接ダウンロードしたサンプルコードやテンプレートを使用している。
・npm（Node.js 系）、pip（Python）、CocoaPods、Gradle、Maven など外部パッケージ管理ツールを使用している。
・GitHub、GitLab 等の公開リポジトリから複製・コピーしたライブラリ・ソースコードを使用している。
・Android Studio や Xcode 等に組み込まれる追加プラグイン/ツールを使用している。
・外部委託会社・サードパーティが組み込んだモジュール・コードがある。

(11) グローバル領域記載定義

VCIドライバのOBD検査用API(ePTIScanForDevices等)のファイル内グローバル領域（インクルード定義からソースコード部の間）に記載する内容を定義する。

(11).1 共通定義

共通定義のフォーマットバージョン(FORMAT_VERSION)について以下の通りとする。

NO	区分	マクロ名	定義する値	備考
1	フォーマットバージョン	FORMAT_VERSION	フォーマットバージョンの値は以下を定義する。 値： ″1.1″	フォーマットバージョンは、グローバル領域記載定義に関する仕様のバージョンを示す。定義項目や内容など仕様が改訂された場合に更新される。

(11).2 メーカー名、VCI デバイス名定義

メーカー名と VCI デバイス名の定義について以下の通りとする。

NO	区分	マクロ名	定義する値	備考
1	メーカー名	VENDOR_NAME	・メーカー名＋VCI デバイス名で 72 文字以下 ・使用可能な文字種類 英字（A～Z、 a～z）	

2	VCI デバイス名	DEVICE_NAME	数字（0～9） 記号：アンダースコア（_）、ハイフン（-）、プラス（+）、カンマ（,）、ピリオド（.）、半角スペース（ ）	
3	メーカーセクション情報	VENDOR_INFO_STR	メーカーセクション情報を生成する定義。 「(11).4 サンプルコード」通りに記載する。	
4	メーカーセクション登録	-	メーカーセクション情報を VCI ドライバに登録するための宣言。 「(11).4 サンプルコード」通りに記載する。	

(11).3 USB 接続定義

USB 接続定義について以下の通りとする。

USB 接続を利用しない VCI デバイスにおいては、USB 接続の定義は不要。

NO	区分	マクロ名	定義する値	備考
1	対応 USB デバイス数	NUM_DEVICE	・VCI ドライバが対応する VCI デバイスのベンダ ID とプロダクト ID の組みわせの数を定義する。 例： ″2″	VCI ドライバが 2 種類のベンダ ID とプロダクト ID の組み合わせに対応する場合″2″と記載する。1 種類の場合は″1″と記載する。
2	ベンダ ID	VENDOR_ID_x	・16 進数型で値前に ″0x″ を付加し、値を 2 バイトで定義する。 例： ″0xFFFF″ ・「x」には対応する USB デバイス数を記載する。（「(11).4 サンプルコード」参照）	USB-IF（USB Implementers Forum）により各メーカーに割り当てられる 2 バイトの識別子を、グローバル領域に定義する。
3	プロダクト ID	PRODUCT_ID_x		ベンダ ID を保有するメーカーが自社製品毎に設定する 2 バイトの識別子を、グローバル領域に定義する。

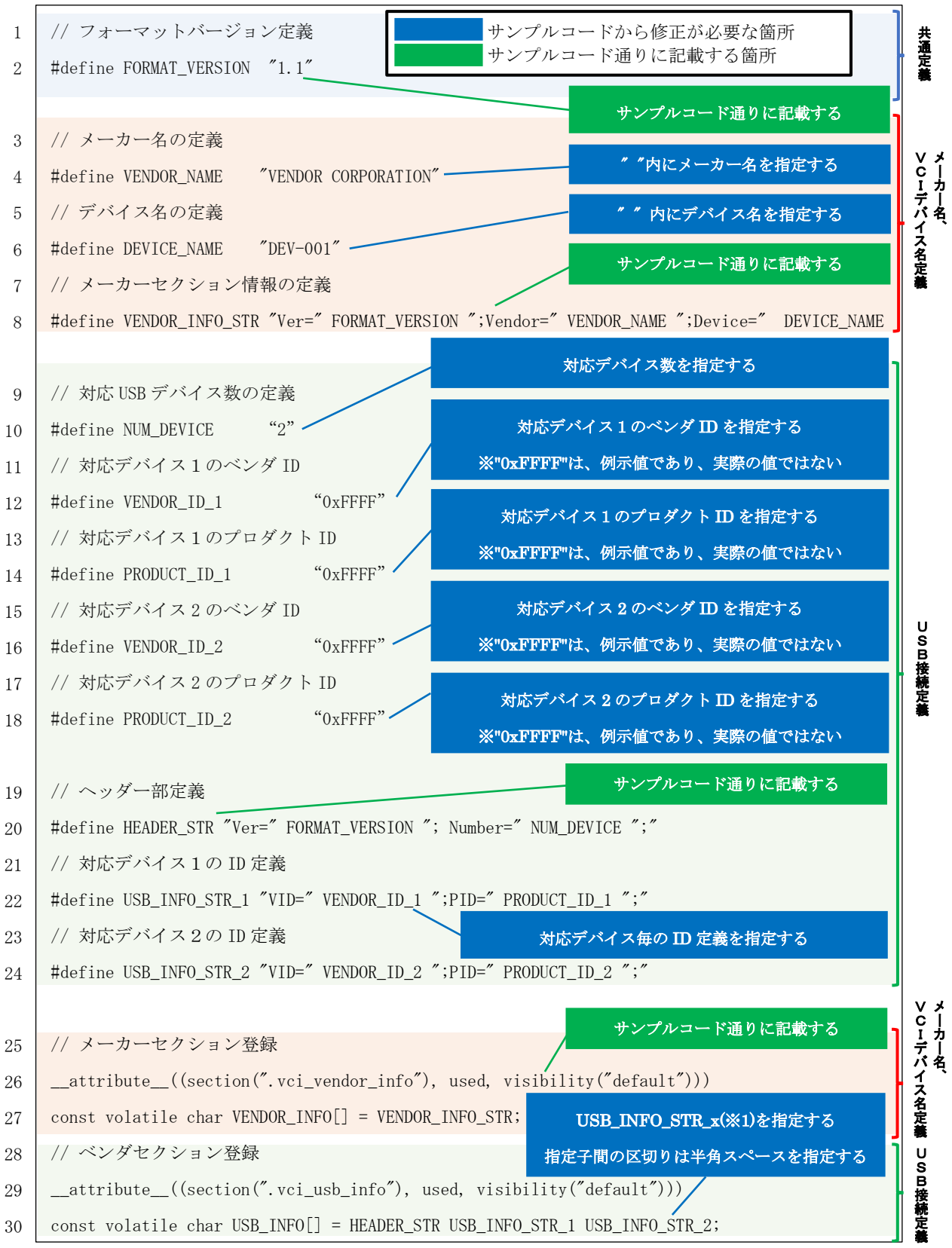
4	ヘッダー部	HEADER_STR	ベンダセクション情報のヘッダー部を生成する定義。 「(11).4 サンプルコード」通りに記載する。	
5	対応デバイスの ID	USB_INFO_STR_x	対応 USB 情報を生成する定義。 ・定義にはデータ判別用のヘッダー情報（例 ″VID=″）とデータ後の区切り情報（″;″）を必ず含む。 ・「x」には対応する USB デバイス数を記載する。（「(11).4 サンプルコード」参照）	
6	ベンダセクション登録	-	ベンダセクション情報を VCI ドライバに登録するための宣言。 USB_INFO_STR_x は対応する USB デバイス数を記載する。（「(11).4 サンプルコード」参照）ベンダセクション情報を VCI ドライバに登録するための宣言。	USB_INFO_STR_x 以外は、「(11).4 サンプルコード」通りに記載する。

(11).4 サンプルコード

サンプルコードを以下に示す。

＜サンプルコード＞

以下に、対応デバイスが 2 件あるケースのサンプルコードを示す。



※USB_INFO_STR_x の値は、デバイス記述子(※2)と整合性を保つものとする。

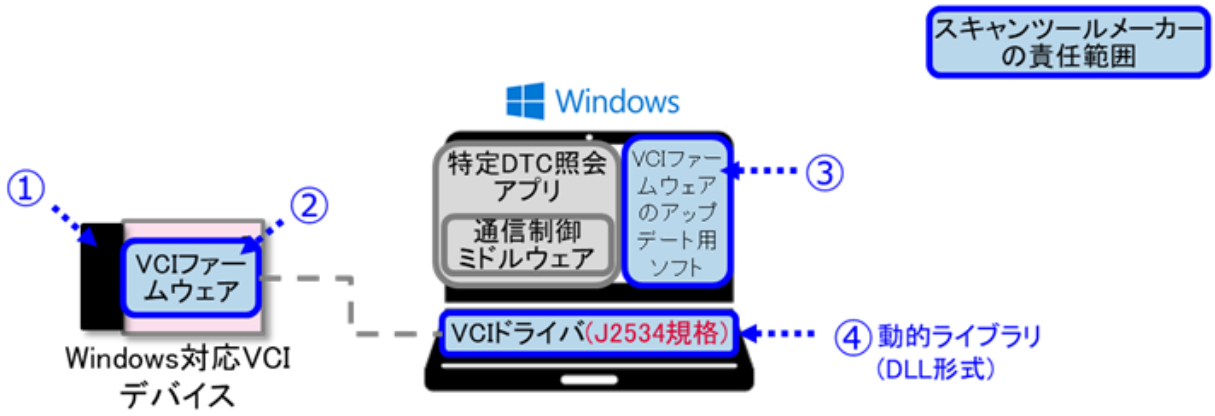
（※1） USB_INFO_STR_x の″ x ″には、定義したベンダ ID とプロダクト ID の数によって記載する USB_INFO_STR_x の数が異なる。サンプルコードの記載は 2 件のケースであり、1 件の場合は、USB_INFO_STR_1 のみの記載となる。

（※2）デバイス記述子とは、USB デバイスの基本情報をホスト（PC など）に伝えるためのデータ構造である。VCI ドライバでは、デバイス記述子のベンダ ID、プロダクト ID に該当する値をグローバル領域に定義することで代替する。

６　スキャンツールメーカーの役割・責任分界点

(1)Windows の場合

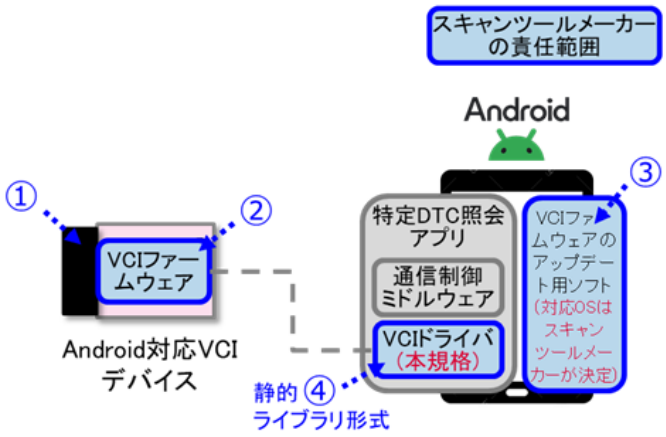
Windows 対応 VCI デバイスを製造・販売するスキャンツールメーカーは、以下を提供するものとする。



#	名称	内容	備考
①	VCIハードウェア	VCIデバイス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デバイス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	スキャンツールメーカー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デバイス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、スキャンツールメーカー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 デバイスを製造・販売するスキャンツールメーカーも、Windows と同様に、以下を提供するものとする。

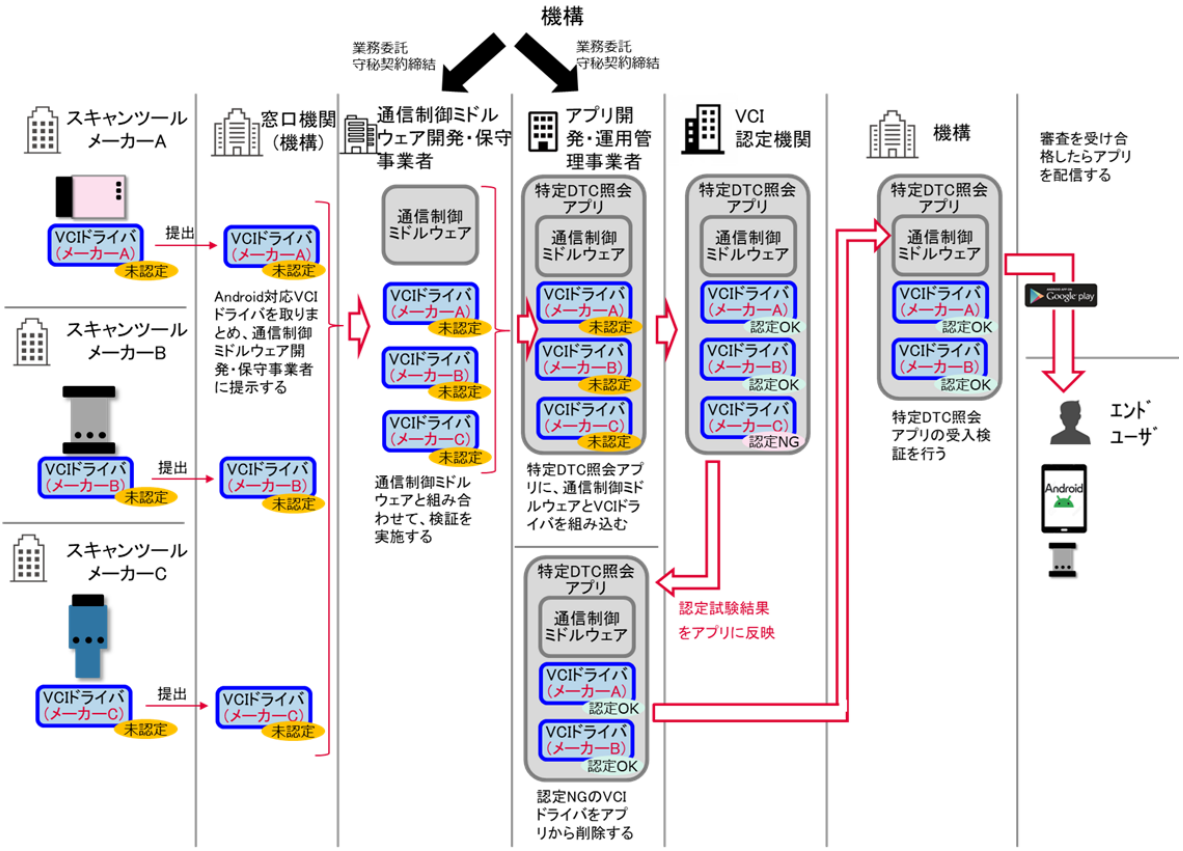


#	名称	内容	備考
①	VCIハードウェア	VCIデバイス本体や接続用ケーブルなどの周辺機器。	
②	VCIファームウェア	VCIの中で動作し、VCIの制御を司る組み込みソフトウェア。	
③	VCIファームウェアのアップデート用ソフトウェア	VCIデバイスのファームウェアをアップデートする時に使用するソフトウェア。	スキャンツールメーカーがエンドユーザに直接配布し、エンドユーザ自身でファームウェアをアップデートする。ただし、本ソフトの対応OSはスキャンツールメーカーが決めるものとする。 (例:Windows版のみの提供でも可とする)
④	VCIドライバ	VCIデバイスを使用してダイアグ通信の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、スキャンツールメーカー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

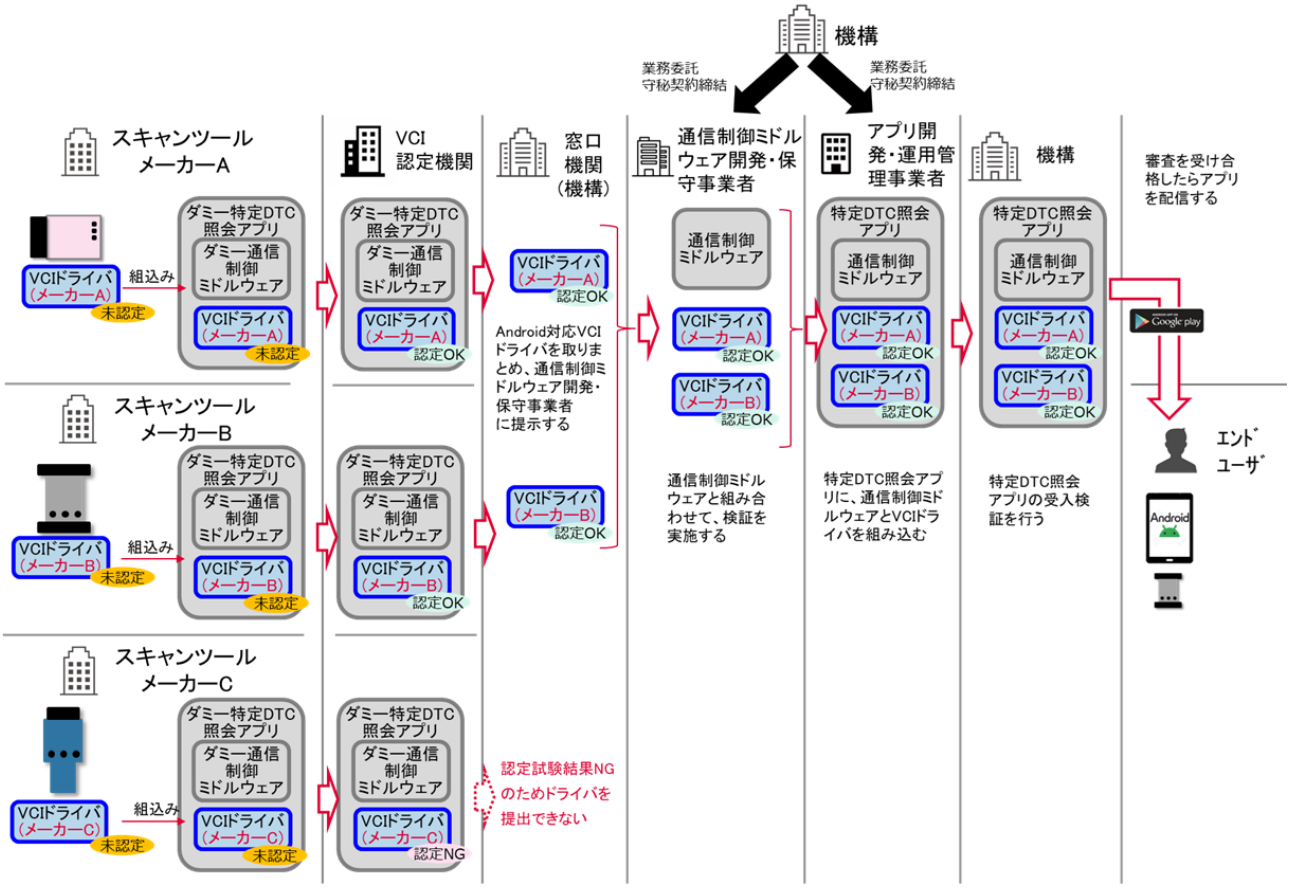
Android 対応 VCI ドライバは、以下のルートで特定 DTC 照会アプリに組み込まれ、エンドユーザに配布されるものとする。このため、VCI ドライバの不具合で修正版 VCI ドライバの配布が必要となった場合、特定 DTC 照会アプリのバージョンアップ計画に合わせてのみ展開されることを基本ルールとする。(ユーザへの影響が大きい不具合の場合は、別途調整するものとする。)

＜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、スキャンツールメーカーがダミーの特定 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する。

＜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、スキャンツールメーカーは評価および品質保証を行うこととする。

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(参考)	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する。 また、事前にMicrosoftから提供される新バージョンの評価版Windowsで検証するものとする。	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する。
Android	バージョンアップ頻度は1年に1回程度 この時のVCIドライバの評価は必須とする。	セキュリティ対応の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

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
	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で検証するものとする。	する。

なお、VCI ドライバの最新バージョンの OS への追従は、スキャンツールメーカーの判断にゆだねるものとするが、追従をしない場合は、窓口機関に報告するものとする。窓口機関から報告を受けたアプリ開発・運用管理事業者は、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する。

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

１０商標について

本書に記載されている会社名、製品名、サービス名は、各社の登録商標または商標です。

Android、Android studio、Google および、Google play は、Google LCC の商標です。

Bluetooth は、Bluetooth SIG, Inc. の登録商標です。

Wi-Fi は、 Wi-Fi Alliance の登録商標です。

Windows は､米国 Microsoft Corporation の米国およびその他の国における登録商標です。

GitHub は、GitHub, Inc. の米国およびその他の国における登録商標です。

GitLab は、GitLab, Inc. の米国およびその他の国における登録商標です。

Xcode は、Apple Inc. の米国およびその他の国における登録商標です。

Java は、Oracle およびその関連会社の登録商標です。

Kotlin は、Kotlin Foundation の登録商標です。

本書では説明の便宜上、商標記号（™、®）を省略しています。