

REVISIÓN

Integration of Alexa Auto and vehicle diagnostic protocols: towards a smart automotive ecosystem

Integración de Alexa Auto y protocolos de diagnóstico vehicular: hacia un ecosistema automotriz inteligente

Alexis Javier Villacis Ninasunta¹, Mariana Pinargote Basurto¹

¹Universidad Técnica Estatal de Quevedo, Facultad de Ingeniería, Departamento de Ingeniería Mecánica. Ecuador.

Citar como: Villacis Ninasunta AJ, Pinargote Basurto M. Integration of Alexa Auto and vehicle diagnostic protocols: towards a smart automotive ecosystem. Transport, Mobility & Society. 2025; 4:210. <https://doi.org/10.56294/tms2025210>

Enviado: 08-08-2024

Revisado: 03-11-2024

Aceptado: 23-07-2025

Publicado: 24-07-2025

Editor: Prof. Emanuel Maldonado 

ABSTRACT

Introduction: the study described how the advancement of digital technologies transformed the automotive industry and drove the adoption of in-cabin voice assistants. It presented Alexa and the Echo Auto device as versatile alternatives to factory solutions, thanks to their synchronization with phones and personalization through skills. He argued that, by integrating with diagnostic standards such as OBDII and the CAN network, such assistants offered immediate access to critical vehicle parameters, reduced driver distraction and facilitated more natural interaction for entertainment, communication and control.

Development: the work reviewed the state of the art and reports that associated the assistants with less visual and manual load and better reaction times. It described integrated ecosystems (MBUX, CarPlay, Android Auto) and external alternatives such as Echo Auto and compatible devices. He detailed the architecture of Alexa and its voice processing pipeline, from audio capture to response synthesis, and presented the development of skills by blocks with Voiceflow and by textual programming through Alexa Skills Kit. He explained the role of IoT platforms, with emphasis on Blynk, to orchestrate data and control devices. Characterized OBDII - connector, protocols and interface - and CAN protocol - topology, speeds, frames and termination requirements - as a basis for extracting and transmitting reliable vehicle data to cloud services and assistance applications.

Conclusions: the research argued that linking Alexa Auto to the in-vehicle computer was feasible and strategic. The convergence between voice assistants, IoT, OBDII and CAN enabled timely diagnostics, hands-free interaction and informed decisions. It was concluded that this integration elevated road safety, optimized car lifespan, and paved the way for smarter vehicle ecosystems and, going forward, an orderly transition to autonomous and sustainable driving.

Keywords: Alexa Auto; OBDII; CAN; IoT; Voice Assistants.

RESUMEN

Introducción: el estudio describió cómo el avance de las tecnologías digitales transformó la industria automotriz e impulsó la adopción de asistentes de voz en cabina. Presentó a Alexa y al dispositivo Echo Auto como alternativas versátiles frente a soluciones de fábrica, gracias a su sincronización con teléfonos y a la personalización mediante skills. Argumentó que, al integrarse con estándares de diagnóstico como OBDII y la red CAN, dichos asistentes ofrecieron acceso inmediato a parámetros críticos del vehículo, disminuyeron la distracción del conductor y facilitaron una interacción más natural para entretenimiento, comunicación y control.

Desarrollo: el trabajo revisó el estado del arte y reportes que asociaron los asistentes con menor carga visual y manual y con mejores tiempos de reacción. Describió ecosistemas integrados (MBUX, CarPlay, Android Auto) y alternativas externas como Echo Auto y dispositivos compatibles. Detalló la arquitectura de Alexa y su canal de procesamiento de voz, desde la captura del audio hasta la síntesis de respuesta, y expuso el

desarrollo de skills por bloques con Voiceflow y por programación textual mediante Alexa Skills Kit. Explicó el papel de plataformas IoT, con énfasis en Blynk, para orquestar datos y controlar dispositivos. Caracterizó OBDII –conector, protocolos e interfaz– y el protocolo CAN –topología, velocidades, tramas y requisitos de terminación– como base para extraer y transmitir datos confiables del vehículo hacia servicios en la nube y aplicaciones de asistencia.

Conclusiones: la investigación sostuvo que enlazar Alexa Auto con el computador del vehículo fue factible y estratégico. La convergencia entre asistentes de voz, IoT, OBDII y CAN habilitó diagnósticos oportunos, interacción manos libres y decisiones informadas. Se concluyó que esta integración elevó la seguridad vial, optimizó la vida útil del automóvil y preparó el camino para ecosistemas vehiculares más inteligentes y, a futuro, para una transición ordenada hacia la conducción autónoma y sostenible.

Palabras clave: Alexa Auto; OBDII; CAN; IoT; Asistentes de Voz.

INTRODUCCIÓN

El vertiginoso avance de las tecnologías digitales ha transformado de manera significativa la industria automotriz, impulsando la incorporación de sistemas inteligentes orientados tanto a la seguridad como al confort del usuario.^(1,2) Entre estas innovaciones destacan los asistentes de voz, cuya adopción ha crecido de forma exponencial en los últimos años, no solo en el ámbito doméstico, sino también en los vehículos. Estas herramientas se han convertido en un elemento clave para la interacción entre el conductor y el sistema de información del automóvil, al reducir la distracción visual y manual, y permitir un acceso más natural y eficiente a funciones de entretenimiento, comunicación y control del vehículo.^(3,4,5)

En este contexto, Amazon ha desarrollado el asistente virtual Alexa y el dispositivo Echo Auto, diseñados específicamente para ofrecer una experiencia optimizada dentro del automóvil. Dichas tecnologías se presentan como una alternativa versátil frente a otros asistentes de voz integrados de fábrica, al permitir la sincronización con dispositivos móviles y la personalización a través de “skills”, que amplían la gama de servicios disponibles.^(6,7) El potencial de estas innovaciones no se limita únicamente al entretenimiento o a la asistencia en tareas básicas, sino que se proyecta hacia la integración con sistemas de diagnóstico vehicular, abriendo un campo de aplicación orientado a la seguridad y el mantenimiento preventivo.^(8,9)

Por otra parte, la introducción del sistema OBDII (On-Board Diagnostics II) y el protocolo de red CAN (Controller Area Network) han sentado las bases para el monitoreo y control de los principales parámetros del vehículo.^(10,11) Estos estándares permiten extraer información en tiempo real acerca del funcionamiento del motor, emisiones y otros sistemas críticos, proporcionando datos fundamentales para prevenir fallas, mejorar la eficiencia y cumplir con normativas ambientales.^(12,13) La posibilidad de enlazar estos datos con asistentes de voz representa una oportunidad para ofrecer al conductor información inmediata y precisa, de manera accesible y no invasiva.^(14,15,16)

El diseño de un sistema de enlace entre Alexa Auto y el computador del vehículo constituye, por tanto, un desafío tecnológico con gran potencial de impacto.^(17,18) Su implementación exige la comprensión del estado del arte en asistentes de voz, la programación y personalización de skills, el uso de plataformas IoT para la administración de dispositivos y el dominio de los protocolos de comunicación automotriz.^(19,20,21) Al integrar estas dimensiones, se plantea un ecosistema capaz de mejorar la experiencia del usuario, incrementar la seguridad vial y optimizar la vida útil del vehículo mediante diagnósticos oportunos y asistencia personalizada.^(22,23,24)

En suma, esta investigación se orienta a fundamentar teóricamente el desarrollo de dicho sistema, explorando las tendencias actuales y las herramientas tecnológicas necesarias para su implementación, con el propósito de ofrecer una solución innovadora que combine inteligencia artificial, conectividad IoT y estándares de diagnóstico automotriz en beneficio del conductor moderno.^(25,26,27)

DESARROLLO

Introducción

En este capítulo se detalla la fundamentación teórica para el diseño e implementación del sistema de enlace entre el computador del vehículo y Alexa Auto, donde se destacan en primer lugar el estado del arte referente a los asistentes de voz y los sistemas de diagnósticos en la industria automotriz.^(28,29,30) De igual manera se abarcan temas acerca del asistente de voz Echo Auto, el desarrollo de skills y las plataformas para la administración de dispositivos IoT.^(31,32) Las cuales son las herramientas necesarias para el desarrollo del sistema.⁽³³⁾

Así mismo se abordan temas relacionados al sistema OBDII, como las interfaces y protocolos para la comunicación, y el sistema de red de área del controlador conocido como CAN, lo cual es la base para lograr una correcta comunicación con el bus de datos bajo el protocolo definido, y así extraer o enviar la información necesaria.⁽³⁴⁾

Estado del Arte

El potencial de la industria automotriz actualmente es el principal factor que impulsa el crecimiento de la inteligencia artificial aplicada en este campo, sobre todo por la preferencia del cliente por funciones nuevas y avanzadas, como asistencia al conductor, conducción autónoma, entretenimiento personalizado, etc. Es por ello que los asistentes de voz se están convirtiendo en una tendencia creciente en los vehículos, además de minimizar la distracción del conductor durante la conducción manual, ofrecen una experiencia cercana a la transición a una conducción autónoma. El panorama actual de asistentes de voz comerciales para vehículos incluye el Asistente de Google a través de Android Auto, Siri de Apple a través de CarPlay, MBUX de Mercedes-Benz, Echo Auto de Amazon Alexa y muchos más.⁽³⁵⁾

Según Smith S.⁽¹⁾, a medida que la tecnología de los asistentes de voz se vuelve más popular en los sistemas automotrices, la necesidad de optimizar estos sistemas se vuelve crucial para una experiencia de conducción sofisticada en el vehículo, ya que, de acuerdo a la investigación, el uso de asistentes de voz se triplicará para 2023. Dentro del campo de la automoción, el 57,6 % de los encuestados en el informe de adopción de asistentes de voz en el vehículo Voicebot de 2019, afirmaron que dentro de un año usarían asistentes de voz mientras estaban en un vehículo con más frecuencia que hasta ese momento. Además, ya el 19,1 % de los consumidores afirma que la asistencia de voz integrada es una consideración importante o incluso un requisito fijo a la hora de comprar un vehículo. De igual manera se ha estudiado que al reemplazar las pantallas tradicionales con asistentes de voz inteligentes da como resultado un mayor rendimiento de conducción, menor tiempo fuera de la carretera, menos desviación de carril y un mejor tiempo de reacción ante accidentes.⁽³⁶⁾

A continuación, se presentan varias aplicaciones llevadas a cabo en el ámbito automotriz, relacionadas con asistentes de voz y sistemas de diagnóstico. Ya que mediante el monitoreo de los sistemas del vehículo se busca mejorar la vida útil del mismo, al alertar al usuario de posibles fallos y por medio de los asistentes de voz el enfoque principal es dar mayor confort a sus usuarios.⁽³⁷⁾

Asistentes de voz y sistemas de diagnósticos en vehículos

Bajo el contexto de mejorar la experiencia del usuario⁽²⁾ hace uso de la información proporcionada por plataformas IoT, para ofrecer soluciones oportunas en la búsqueda de lugares de aparcamiento para vehículos utilizando un asistente de voz inteligente. De este modo el autor realiza una skill para Alexa, que se conectará con la plataforma FIWARE; la cual recolecta datos del entorno proporcionados por los usuarios, como sensores e incluso aplicaciones móviles. Esto permite realizar una petición al asistente virtual y que este genere una respuesta que lo guiará hasta un lugar de aparcamiento disponible.⁽³⁸⁾

En el trabajo realizado por Caicedo et al.⁽³⁾ se busca identificar oportunamente fallas o errores que pongan en riesgo el funcionamiento del vehículo. Con el objetivo de preservar la vida útil del vehículo y mantener informado al conductor del estado del vehículo en tiempo real.⁽³⁹⁾ Esta información es presentada en interfaces WEB, donde el usuario pueda revisar tal información con mayor detalle. En cuanto al hardware utilizado, este sistema requiere de la conexión del módulo de diagnóstico ELM327 al puerto OBDII del vehículo, y vía bluetooth se envían los datos recolectados a un microcontrolador ESP32, que se encarga de la decodificación de los mismo. A su vez, el ESP32 está conectado a un computador (LattePanda), donde se envían los datos a la nube para que puedan ser monitoreados en una interfaz más detallada.⁽⁴⁰⁾

Asistentes de voz para vehículos

Existen dos alternativas principales en las que se puede encontrar un asistente de voz en un vehículo: asistentes de voz integrados desde la fabricación del vehículo y asistentes de voz con dispositivos independientes. Dependiendo de la que se encuentre disponible en el vehículo, el usuario podrá acceder a diferentes funcionalidades relacionadas al entretenimiento y multimedia o incluso del control del vehículo.⁽⁴¹⁾

Las opciones que vienen integradas de fábrica pueden ser sistemas propios de las marcas del vehículo, tales como Mercedes-Benz con su asistente MBUX, BMW con BMW Intelligent Personal Assistant o Audi con su sistema operativo MMI Touch Response. Otra opción de integración disponible se debe a que gran variedad de fabricantes han realizado convenios con Amazon, Google o Apple, para instalar sistemas compatibles con Android o iOS, y utilizar un smartphone para sincronizarse con su centro de información y entretenimiento. La ventaja principal de estos sistemas es que permiten acceder a funciones adicionales de control del vehículo. Tales como, arranque del motor, control de temperatura, seguro de puertas y entre otras que buscan hacer más comfortable la estadía del conductor.⁽⁴²⁾

El sistema operativo CarPlay de Apple, posibilita presentar la pantalla de un iPhone en la pantalla digital del tablero del vehículo. Permitiendo al usuario acceder a las funciones del teléfono de manera mucho más segura y sin la necesidad de quitar las manos del volante, incluso interactuar con el asistente de voz Siri. Actualmente una gran variedad de vehículos lo han incluido o son compatibles con este sistema. Los modelos y marcas se pueden verificar en la página oficial de Apple.⁽⁴⁾

Por otra parte, Google cuenta con Android Auto que es una aplicación móvil que permite conectar un teléfono

Android a la pantalla del vehículo. Para que el usuario pueda acceder a las aplicaciones de su smartphone por comandos de voz. Los modelos más nuevos de los principales fabricantes, incluyendo Ford, Nissan, Toyota, Mazda, Hyundai, etc., son compatibles con Android Auto.⁽⁵⁾

En el caso de Amazon, este ha realizado diversos convenios con varias marcas para incluir Alexa desde la fabricación de automóviles, lo que permitirá controlar el vehículo desde cualquier lugar que el usuario lo desee o que cuente con un dispositivo Echo. Actualmente una gran variedad de marcas y modelos cuentan con esta tecnología, estos se pueden verificar en su página oficial; donde incluso se encuentra disponible una skill específica para cada uno.⁽⁶⁾

En cuanto a las opciones para vehículos que no tengan esta tecnología integrada, existen opciones de aplicaciones móviles como AutoMate que permiten interactuar con el smartphone por comandos de voz, sin embargo, sus funciones son limitadas a aplicaciones de navegación, música, telefonía y mensajería. Otras opciones que requieren de dispositivos físicos como JBL Link Driv o Anker Roav Bolt permiten al usuario conectarse con Google Assistant y a sus funcionalidades. O también pueden utilizar un dispositivo Echo Auto para interactuar con el asistente de voz Alexa de Amazon.⁽⁷⁾

Asistente de voz Alexa y Echo Auto

Alexa es un asistente virtual desarrollado por Amazon, el cual se utiliza en dispositivos como los altavoces inteligentes Echo. Su principal función es proporcionar una interfaz de usuario basada en voz para que las personas puedan interactuar con otros dispositivos o servicios. Entre las funciones más comunes para este asistente se encuentran el control de dispositivos inteligentes, acceso a servicios de entrenamiento, información general a través de búsquedas en Internet, creación de listas o recordatorios, servicios de llamadas y mensajería, así como también habilidades personalizadas.⁽⁴³⁾

Cada vez que el usuario realiza una petición, se lleva a cabo una serie de procesos que permiten al asistente de voz interpretar el mensaje dado por el usuario y dar una respuesta adecuada. Para esto, en la figura 1, se muestra un proceso de 7 pasos principales, que describen el flujo de procesamiento que se lleva a cabo:

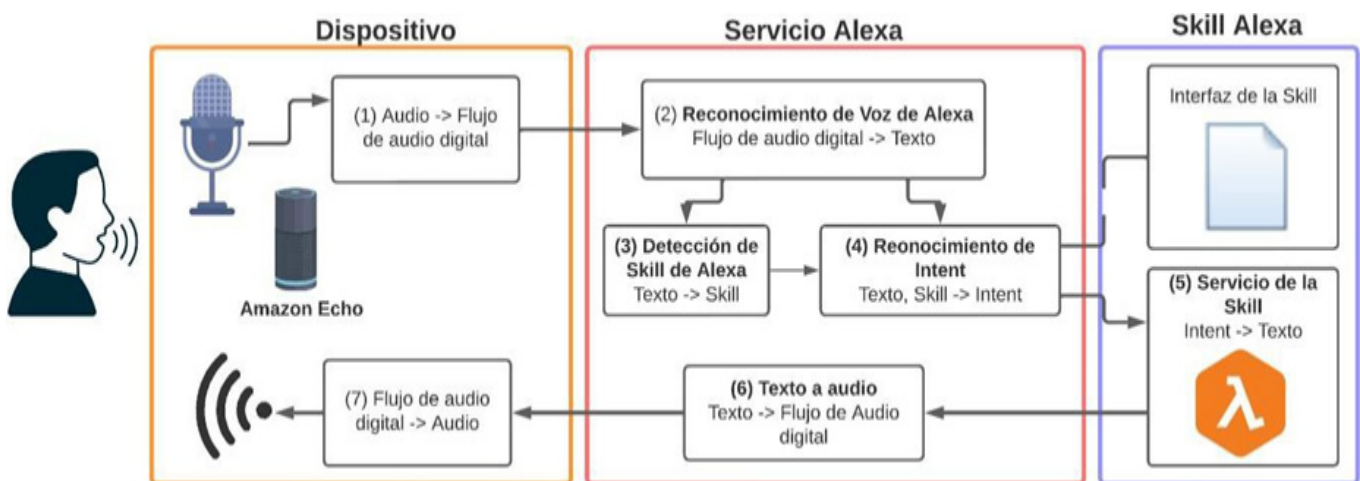


Figura 1. Canal de procesamiento de Alexa

- Conversión de audio: el dispositivo físico se encarga de capturar el audio, digitalizarlo, encriptarlo y enviar como un flujo digital a la nube de procesamiento de Alexa.
- Reconocimiento de voz: permite la transformación del flujo de audio digital en una cadena de texto, para lo cual se utilizan servicios correspondientes a Speech to Text.
- Detección de la Skill: el texto reconocido permite a Alexa determinar la aplicación correspondiente para activarla.
- Reconocimiento de la Intención: Alexa utiliza el modelo de reconocimiento de intenciones de la skill respectiva y extrae la petición del usuario junto con otros parámetros importantes del texto.
- Servicios de la Skill: para generar una respuesta adecuada, Alexa accede a servicios que reciben la información extraída previamente del texto para generar respuestas textuales a lo que solicita el usuario.
- Conversión de texto a flujo de audio: se utilizan los servicios de Text to Speech de Alexa para generar un flujo de audio digital de la respuesta textual y enviarlo al dispositivo Echo.
- Conversión de flujo de audio: el dispositivo Echo se encarga de reproducir el audio generado para dar una respuesta audible a la petición del usuario.

En cuanto a los dispositivos Echo de Alexa, son una variedad de altavoces inteligentes que se controlan por voz. Estos pueden incluir el asistente de voz Alexa de fábrica o pueden requerir de un smartphone con la aplicación móvil de Amazon Alexa para acceder al asistente.⁽⁴⁴⁾ Un caso específico de estos dispositivos es el Echo Auto, que está diseñado específicamente para ser instalado dentro de un vehículo. Por lo cual, sus características físicas están construidas para funcionar en estos ambientes propensos a ruidos constantes y con un tamaño compacto para no obstruir otras partes del vehículo. Este permite utilizar funcionalidades de manos libres de Alexa, tales como, reproducir música, hacer llamadas, acceso a búsquedas en internet y muchas otras funciones para interactuar con el usuario sin que este quite las manos del volante.⁽⁸⁾

Skills de Alexa

Alexa es el servicio de voz ubicado en la nube de Amazon disponible en dispositivos propios de la marca y dispositivos terciarios con Alexa integrada. Además, cuenta con un sin número de funcionalidades, a lo que Amazon llama «Skills», que permiten a los consumidores crear una experiencia más personalizada.⁽⁹⁾

Para acceder al contenido de una skill, se realiza una solicitud al dispositivo o plataforma Alexa para invocarla. Al hablar de invocar se refiere al acto de comenzar una interacción con una skill particular de Alexa, en la figura 2, se presenta la estructura básica de una solicitud para interactuar con el asistente de voz.⁽⁴⁵⁾



Figura 2. Estructura básica de una petición

Como detalla Domínguez⁽¹⁰⁾ el primer paso, implica nombrar la palabra de activación (o en inglés, wake word), que generalmente es “Alexa”, aunque es personalizable con opciones como “Echo”, “Amazon” u otra opción. Enseguida, se emplea la palabra de lanzamiento (o en inglés, launch), que indica a Alexa la acción que desea el usuario; entre las alternativas se encuentran inicia, abre, comienza, pídele a, o pregunta.⁽⁴⁶⁾

A continuación, se menciona el nombre de invocación (o en inglés, invocation name), con el cual los usuarios se referirán a la skill específica, es importante que se seleccione adecuadamente el nombre de la misma, para evitar errores de comprensión del asistente de voz. Para complementar la información solicitada por el usuario es esencial agregar detalles adicionales conocidos como enunciado (o en inglés, utterance), aquello sirve para definir concretamente la ruta de acción de la skill.⁽¹⁰⁾

Plataformas de Programación de skills para Alexa

Según un estudio con la creciente popularidad de los asistentes de voz existen plataformas proporcionadas por el fabricante de los dispositivos de Alexa, así como diversas plataformas de empresas independientes que permiten desarrollar skills mediante programación por código (JavaScript o Python), bloques o plantillas. A continuación, se detallan algunos ejemplos de dichos programas:

- Plataformas proporcionadas por Amazon Alexa o Alexa Skills Kit (ASK): Es un conjunto de herramientas desarrolladas por Amazon para el diseño de asistentes de voz, específicamente para la creación de skills de Alexa, que incluye la documentación oficial, ejemplos de código y una interfaz de programación de aplicaciones (API).⁽¹¹⁾
 1. Blueprints: esta plataforma, desarrollada por Amazon, permite la creación de skills personalizadas sin necesidad de codificación, ya que ofrece plantillas predefinidas que pueden ser personalizadas con acciones, preguntas y respuestas, programar tareas y realizar seguimientos.⁽¹²⁾
- Plataformas de empresas independientes
 2. Voiceflow: es una herramienta para crear y diseñar conversaciones e interacciones de skills de voz a partir de programación por bloques. Mediante la creación de flujos de conversación proporciona una interfaz que es compatible con varios asistentes de voz, incluyendo Alexa, Google Assistant y entre otros.⁽¹³⁾
 3. Dialogflow: es una plataforma que incorpora tecnología de comprensión del lenguaje natural, ya sea a través de texto o audio de las personas, facilitando la creación de interfaces de usuario conversacionales en diversas aplicaciones, como dispositivos móviles, aplicaciones web, bots y asistentes de voz con Alexa.⁽¹⁴⁾

Tipos de programación para skills

Para la creación de skills de Alexa, se pueden clasificar en términos generales dos tipos de programación: la programación por bloques y la programación textual. En este contexto, los enfoques de desarrollo para ambas

modalidades se definirán en plataformas particulares; por ejemplo, Voiceflow para la programación por bloques y Alexa Skill Kit para la programación textual.

Programación por bloques en Voiceflow

La programación por bloques, también conocida como programación visual, ofrece la comprensión del flujo de las conversaciones de voz permitiendo incorporar una rápida iteración y creatividad a lo largo del diseño de asistentes de voz.⁽¹⁵⁾ La plataforma Voiceflow permite la creación de skills de Alexa, mediante programación por bloques. En esta se puede arrastrar y soltar bloques de funciones para definir los flujos de conversación y crear prototipos rápidos de skills.⁽¹⁶⁾ Algunas de las características de esta herramienta es la integración con información de otras plataformas IoT, dispone de una interfaz para verificar la funcionalidad de la skill y permite publicar directamente en la consola de desarrollo de Alexa Skills Kit (Alexa Developer Console).⁽¹³⁾ La interfaz de Voiceflow contiene múltiples bloques para desarrollar la programación de la skill como se observa en la figura 3.

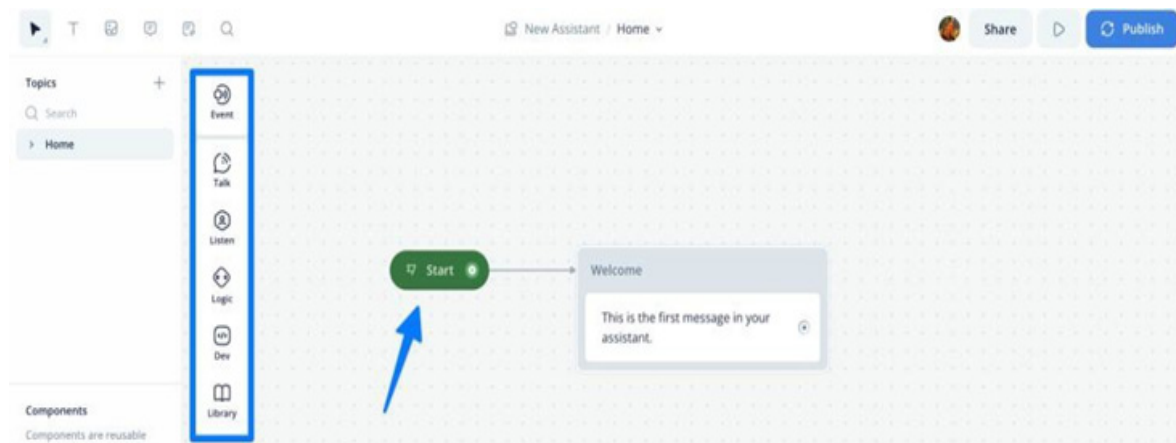


Figura 3. Interfaz de Voiceflow y sus componentes básicos⁽¹⁷⁾

Como explica Tahsim⁽¹⁸⁾, los bloques representan las distintas funcionalidades que puede incorporarse a la estructura de la conversación, se clasifican en:

- Bloque Evento (Event): representa el inicio del flujo de conversación de la skill, donde se puede configurar las acciones iniciales y la intención principal de la asistencia de voz.
- Bloque Hablar (Talk): permite la emisión de respuestas de voz para el usuario, entre las opciones de respuestas del bloque se puede transformar el texto en audio, transmitir un audio, visualizar una imagen o video, y finalmente proyectar archivos multimedia.
- Bloque Escuchar (Listen): captura la entrada del usuario, ya sea voz o texto, para guiarlo a través del flujo de conversación establecido. Esto se logra mediante la identificación de comandos almacenados en la base de datos de la plataforma, donde se establecen dos caminos para el flujo de la conversación: el bloque de elección de alternativas para dirigir a un resultado predeterminado y el bloque de captura, donde se registra información proporcionada por el usuario.
- Bloque Lógica (Logic): controla la toma de decisiones en la programación del asistente y almacena datos pertinentes como condiciones, variables, bucles, finalizar el código, etc.
- Bloque Desarrollador (Dev): habilita fragmentos de código que permiten la ejecución y simulación de la skill en base a las condiciones establecidas, facilitando la verificación de errores.
- Bloque Biblioteca (Library): facilita el acceso a plantillas de bloques para que se puedan editar y personalizar a las necesidades del usuario.

Cabe recalcar que esta plataforma puede conectarse con servicios externos a través de bloques API, los cuales hacen posible el acceso a variables de dispositivos o sitios web externos. Este bloque consta de varias partes como se indica en la figura 4:

1) Tipo de solicitud: son solicitudes que implican recuperar datos de una base de datos externa o enviar datos a otra fuente, entre otras opciones. Hay cinco solicitudes disponibles:

- GET: Recupera información.
- POST: Crea información.
- PUT: Actualiza completamente la información.
- DELETE: Elimina información.
- PATCH: Actualiza parcialmente la información.

- 2) Endpoint: se refiere al código URL o la variable en la que se establece o se solicita el dato de la API.
- 3) Encabezado, cuerpo y parámetro: es la información adicional para acceder u orientar a la API como nombres de usuario, contraseñas, tokens de autenticación, etc.
- 4) Captura de respuesta / Guardar Variables: guarda la información obtenida de la API, esto permite establecerla como respuesta para continuar el flujo de la conversación.
- 5) Envío de solicitud: permite verificar la solicitud de la API al sitio externo requerido.

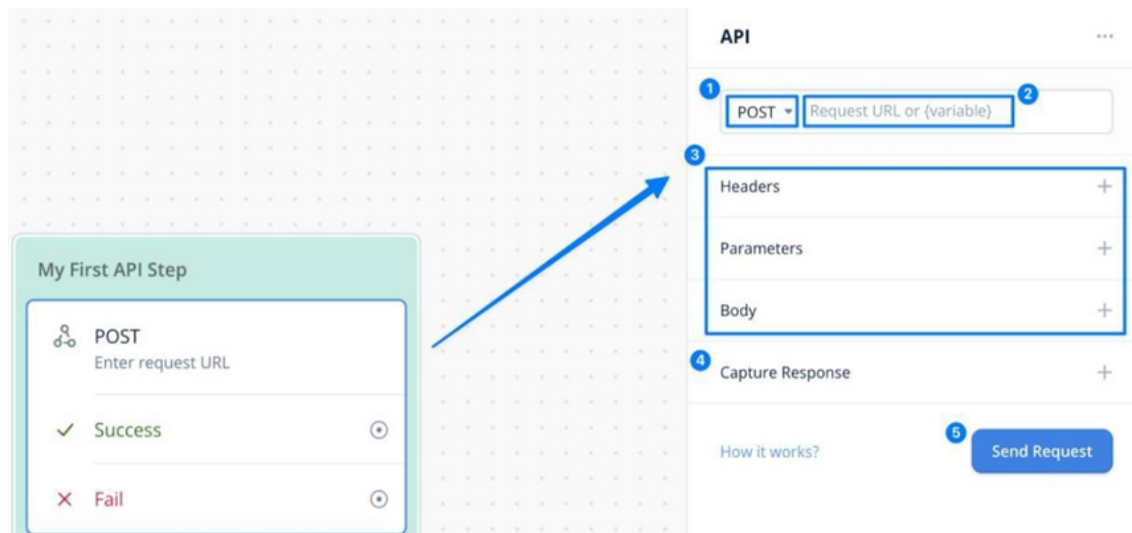


Figura 4. Partes del Bloque API⁽¹⁷⁾

Programación textual en Alexa Skills Kit

Este tipo de programación es fundamental en el avance de los asistentes de voz, brindando a los desarrolladores la capacidad de concebir comandos y respuestas personalizadas. La programación textual implica la utilización de lenguajes de programación como JavaScript o Python, para la creación de habilidades personalizadas destinadas a Alexa.⁽¹⁵⁾ En este contexto, Amazon tiene a disposición la plataforma Alexa Skills Kit (ASK), la cual integra herramientas, documentación, ejemplos de código fuente y API diseñados para la creación de skills específicas para Alexa. Como se visualiza en la figura 5, el flujo para el desarrollo de skills de Alexa tiene la siguiente estructura:

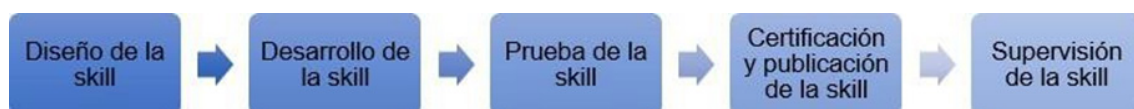


Figura 5. Flujo de desarrollo de una skill de Alexa⁽¹⁹⁾

```

const LaunchRequestHandler = {
  canHandle(handlerInput) {
    return Alexa.getRequestType(handlerInput.requestEnvelope) === 'LaunchRequest';
  },
  handle(handlerInput) {
    const speechText = 'Welcome to your SDK weather skill. Ask me the weather!';

    return handlerInput.responseBuilder
      .speak(speechText)
      .reprompt(speechText)
      .withSimpleCard('Welcome to your SDK weather skill. Ask me the weather!', speechText)
      .getResponse();
  }
};
  
```

Figura 6. Ejemplo del controlador para la solicitud de lanzamiento de una skill

A continuación, se detallan ciertos aspectos de cada fase del proceso de desarrollo de la skill:

- **Diseño:** en esta fase, se selecciona el modelo de interacción por voz, se definen las palabras y frases que los usuarios pueden emplear para activar la skill, se incluyen aspectos visuales y la posibilidad de incorporar aplicaciones externas, así como otros elementos.⁽²⁰⁾
- **Desarrollo:** en esta etapa, es posible desarrollar una skill de dos maneras distintas. En la primera opción, se emplea una plantilla o modelos prediseñados en ASK, donde se configura el modelo de interacción, intenciones y ejemplos de expresiones para personalizar la habilidad.⁽²⁰⁾ La segunda alternativa también se desarrolla en la misma plataforma ASK, sin embargo, se puede personalizar mediante programación textual usando lenguajes como JavaScript, Python, C, Go, Ruby, entre otras.⁽²⁰⁾ Para manejar las solicitudes de los usuarios, se basa en la lógica de programación de dos funciones fundamentales: `canHandle` y `handle`, las cuales se encuentran en los controladores para cada intención de la skill.

En la función `canHandle`, se establecen las condiciones o criterios que deben cumplirse para que el controlador asuma la responsabilidad de manejar una solicitud en particular.

La función `handle` tiene la tarea de generar una respuesta al usuario. Cuando la habilidad recibe una solicitud, la función `canHandle` dentro de cada controlador evalúa si ese controlador específico es apto para atender la solicitud en función de las condiciones establecidas previamente. Si se cumple la condición, el controlador sume el manejo de la solicitud y la función `handle` se activa para generar la respuesta correspondiente que será comunicada al usuario.

Como se mencionó anteriormente, existen controladores que gestionan el flujo de la skill. Estos controladores posibilitan la definición de las intenciones como los comandos de inicio, la solicitud de variables, la obtención de información necesaria, la petición de ayuda o detalles específicos, así como la finalización de la interacción con la skill. La figura 6 presenta un ejemplo de código que configura el controlador para la solicitud de lanzamiento. Este controlador tiene la función de iniciar la skill, cuando la skill recibe la invocación de activación por parte del usuario, y se detecta el comando `LaunchRequest`, lo cual desencadena la ejecución de la skill.

- **Prueba:** en esta sección, se dispone de un entorno de pruebas para la verificación de errores que incorpora un simulador de Alexa con capacidades de salida en texto y audio.
- **Certificación y publicación:** una vez confirmado el correcto desempeño de la skill, se procede a su certificación en la plataforma de Amazon. Donde se verifica que la aplicación cumpla con los requisitos establecidos en términos de políticas, seguridad, funcionalidad, así como en la interfaz de voz y la experiencia del usuario. Luego, se procede a la publicación de la skill en la tienda oficial de Amazon.
- **Supervisión:** finalmente, en esta fase, se posibilita la supervisión del uso, la realización de análisis, visualización de los ingresos generados por la skill, y entre otras variables relevantes.

Plataformas para administración de dispositivos IoT

Las plataformas IoT permiten administrar información de dispositivos conectados a internet, de modo que se puedan controlar y gestionar de forma remota. Debido a la complejidad y variedad de los dispositivos, se pueden encontrar diversas categorías de plataformas IoT dependiendo de las aplicaciones para las que estén destinadas. Entre las más comunes se pueden destacar plataformas de habilitación de aplicaciones IoT, de conectividad, de dispositivos, de análisis de datos y entre otras.

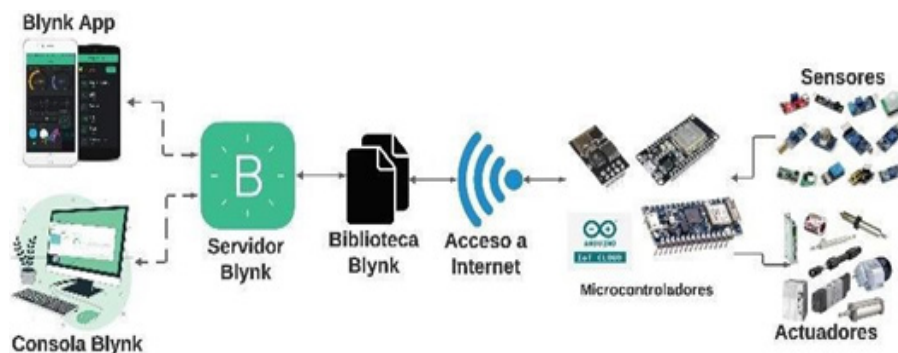


Figura 7. Integración de la plataforma Blynk con dispositivos externos

Debido a la gran variedad, la selección de una plataforma adecuada dependerá de diversos factores tales como el campo de aplicación, necesidades técnicas, operativas o el alcance al que se dese llegar. Como ejemplo de las más conocidas actualmente, se pueden destacar: Google Cloud Platform, Particle, Amazon AWA IoTCore, Azure IoT Hub, NodeRed, etc. Blynk, es otro ejemplo de plataformas IoT, la cual se caracteriza por permitir a los usuarios conectar hardware a la nube y crear aplicaciones para Android, iOS y web. Entre sus funcionalidades se destaca la posibilidad de controlar y gestionar dispositivos electrónicos, así como también el análisis de datos

históricos en tiempo real.⁽²¹⁾ Cuenta con un componente importante denominado Blynk.Edgent para facilitar la conexión de dispositivos compatibles con la plataforma, entre los cuales se incluyen microcontroladores como ESP32, ESP8266, varios modelos de Arduino con conexión a Internet, Raspberry y entre otras. Esto se logra gracias a la biblioteca Blynk, que permite comunicar el dispositivo con la plataforma de manera bidireccional. Como se muestra en la figura 7, esta plataforma permite la conexión con los dispositivos de manera sencilla y su monitorización o control ya sea desde aplicaciones móviles o web.

Estándar de diagnóstico OBDII

OBD es un sistema de diagnóstico a bordo para vehículos. La compañía General Motors fue el primer fabricante en incorporar un sistema de diagnóstico en sus modelos de la década de 1980, principalmente para monitorear sistemas de inyección y hacer ajustes simples, y así cumplir con los estándares de emisiones de la Agencia de Protección Ambiental (EPA, Environmental Protection Agency). Este primer sistema fue denominado ALDL (Assembly Line Diagnostic Link) o Enlace de diagnóstico de línea de montaje.

Para 1988, el Consejo de Recursos del Aire de California determinó que todos los vehículos propulsados por gasolina deben estar equipados con alguna capacidad del sistema OBD básico; conocido como OBD-I, que utiliza dispositivos electrónicos para monitorear y hacer cumplir los límites de emisiones de gases de los vehículos. Con la mejora de los sistemas de diagnóstico a bordo y los avances tecnológicos, se desarrolló una nueva generación del sistema OBD, que se instauró a partir de 1996 de manera obligatoria para vehículos comercializados en Estados Unidos. Este fue conocido como OBDII, que se instaló en la mayor parte de automóviles y vehículos industriales, para monitorear fallos que tengan relación con la emisión de gases de escape del vehículo.

Debido a las diferencias en las regulaciones, requisitos de emisiones y estándares regionales en todo el mundo, el estándar OBD cuenta con diversas variaciones y extensiones que buscan el cumplimiento de requerimientos específicos ya sea a nivel local o global. Por ejemplo, la variante de Diagnóstico a Bordo para Europa (EOBD, European On-Board Diagnostics) adapta el estándar OBDII para que los vehículos fabricados y comercializados en Europa cumplan con las regulaciones de la región. La norma SAE J1979 define una serie de variantes del estándar OBD, con las que se puede diseñar la Unidad de Control del Motor (ECU, Engine Control Unit) para que cumpla las especificaciones referentes a las emisiones permitidas en una región específica.

El diagnóstico del vehículo se realiza en torno a los principales parámetros relacionados con el correcto funcionamiento del motor; tales como la velocidad, carga, temperatura del motor, consumo de combustible, temperatura del ambiente, caudal de aire y emisiones de gases. Todos los sensores relacionados se conectan a la ECU, la cual se encarga de controlar todas las funciones para que el motor opere adecuadamente en función de los valores registrados por sus sensores. Si alguno de los parámetros principales se encuentra fuera de los rangos establecidos, el sistema OBDII se encarga de almacenar y procesar esta información para alertar al conductor de un mal funcionamiento.

Actualmente, aún está en desarrollo una versión mejorada de este sistema de diagnóstico que se denominará OBDIII, con la cual se busca modernizar el diagnóstico de los vehículos. Con la capacidad de detectar cualquier tipo de fallas y enviar toda esta información a una agencia reguladora para controlarlas. La comunicación de cualquier dato del vehículo a distancia se pretende realizar vía satélite, de modo que en cuanto se encienda la luz de mal funcionamiento del vehículo podrá ser detectado y reportado vía satélite en cualquier parte del planeta.⁽²²⁾

El protocolo OBDII se desarrolla bajo un conjunto de normas para facilitar principalmente el diagnóstico de las emisiones de un vehículo. La ISO 15031-1 realiza una introducción sobre las normas internacionales que forman parte de las especificaciones para los sistemas de diagnóstico, y las asigna según el modelo de Sistemas Abiertos de Interconexión (OSI, Open System Interconnection). En Tabla 1, se detalla cada una de las normas ISO en concordancia con el modelo OSI de 7 capas. El uso de estas estas normativas está basado en la SAE J1962, la cual especifica las características físicas del conector de diagnóstico utilizado.

Tabla 1. Especificaciones de diagnóstico relacionadas con las emisiones aplicadas al modelo OSI				
Modelo OSI - 7 Capas		Estándares aplicados		
Capa 7 - Aplicación		ISO 15031-5		
Capa 6 - Presentación		ISO 15031-2, 5, 6 SAE J1930-DA / SAE J1979-DA SAE J2012-DA (OBD)		
Capa 5 - Sesión	No aplica		ISO 14229-2	
Capa 4 - Transporte	ISO 15031-5	ISO 14230-4	ISO 15765-2	ISO15765-4
Capa 3 - Red				
Capa 2 - Enlace de datos	SAE J1850	ISO 9141-2	ISO 14230-2	ISO 11898-1
Capa 1 - Física		ISO 14230-1	ISO 11898 - 1, 2	

Interfaces para comunicación OBDII

El conector estándar de OBDII es conocido como Conector de Enlace de Diagnóstico (DLC, Diagnostic Link Conector). Normalmente se encuentra ubicado bajo el tablero o junto a este, aunque su ubicación puede variar según sea el modelo o marca del vehículo. Según las especificaciones de la SAE J1962 existen dos interfaces de hardware estándar, de 16 pines. Estas corresponden a los tipos A y B, como se muestra en la figura 8, el tipo A tiene una ranura continua en medio de las dos filas de pines, mientras que el tipo B tiene una ranura interrumpida en el medio. Además, el tipo A se utiliza para vehículos que tienen una tensión de alimentación de 12V y el tipo B se utiliza para alimentaciones de 24V con un color azul distintivo en el conector.

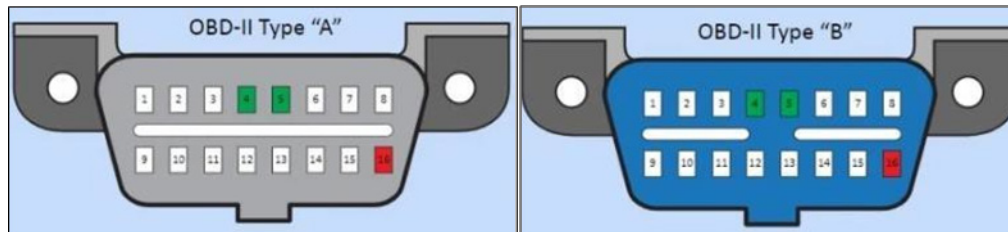


Figura 8. Interfaces para OBDII especificadas por la SAE J1962

Se detalla cada uno de los pines con los que cuenta el conector OBDII, tanto para el tipo A como para el tipo B. El uso de estos pines dependerá del fabricante y de las necesidades de comunicación del vehículo. Los sistemas OBDII cuentan con 5 protocolos de comunicación, y cada fabricante selecciona uno solo para implementarlo en el vehículo.

Se puede identificar el tipo de protocolo que posee el vehículo según los pines que tengan cobre o material conductor en el pin específico. En la figura 9, se muestran los pines que al menos debería tener el puerto según el protocolo, ya que aparte de los que se resaltan el fabricante puede optar por pines adicionales, ya sea para alimentación o con fines no relacionados al protocolo, como puede ser el caso para dispositivos de diagnóstico específicos de la marca:

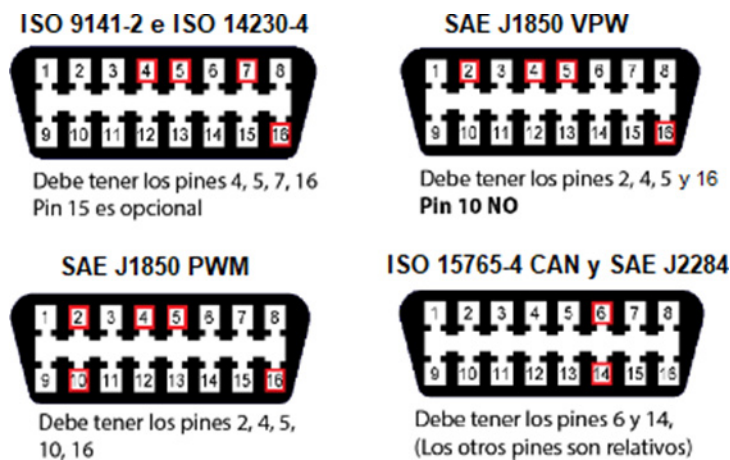


Figura 9. Conectores para protocolos OBDII⁽²³⁾

Protocolos de comunicación

A partir de 2008 en Estados Unidos el protocolo de comunicación CAN se volvió obligatorio para todos los vehículos fabricados, sin embargo, es común encontrar en circulación modelos fabricados previo a 2008 o de algún otro país de origen, con alguno de los otros protocolos que soporta OBDII.⁽²⁴⁾ A continuación, se describen estos cinco protocolos:

- ISO 9141-2: protocolo utilizado en vehículos europeos, la marca Chrysler y algunos asiáticos, entre 2000 y 2004. Consiste en una comunicación serie asíncrona a una velocidad de 10,4 kbps. Utiliza obligatoriamente los pines 4, 5 y 16 del conector de OBDII, mientras que los pines 7 y 15 son opcionales.
- ISO 14230-4 (KWP2000): se refiere al Protocolo de Palabras Clave muy común en vehículos europeos y de la marca Chrysler. Requiere obligatoriamente los pines 4, 5, 7 y 16, y el pin 15 es opcional. Admite varias velocidades de transmisión de datos, siendo los más comunes a 5 kbps y 10,4 kbps.
- SAE J1850 PWM: protocolo de Modulación por Ancho de Pulso, permite una comunicación de hasta 41,6 kbps en modo diferencial sobre dos cables. Requiere de los pines 2 y 10 para la comunicación, y los pines 4, 5 y 16 para alimentación. Es utilizado en principalmente en los modelos más antiguos de Ford y otras marcas estadounidenses.

- SAE J1850 VPW: protocolo de Modulación por Ancho de Pulso Variable, usado principalmente en vehículos de General Motors. Opera a una velocidad de transmisión de 10,4 kbps sobre un único cable referido a masa. Requiere de los pines 2, 4, 5 y 16.
- ISO 15765-4 CAN (SAE J2480 CAN) y SAE J2284: protocolo CAN BUS, utilizado desde 2003 y a partir de 2008 es obligatorio en vehículos estadounidenses. Los pines 6 y 14 son obligatorios para el protocolo mientras que los pines 4, 5 y 16 se utilizan para alimentación. Permite diferentes velocidades de transmisión de datos hasta 1 Mbps, en modo diferencial sobre dos cables.

Sistema de Red de Área del Controlador (CAN)

El protocolo de comunicación CAN fue diseñado por la compañía de Robert Boch, en 1986, para ser implementado en la industria automotriz. Años más tarde, fue estandarizado por sus ventajas, tales como: la robustez del sistema, las altas velocidades de transmisión, la reducción del cableado, la flexibilidad en la configuración del bus, entre otras. Este sistema de comunicación serie requiere de dos cables (CAN-H y CAN-L), para conseguir una señal diferencial, con una velocidad de transmisión de hasta 1Mbps.

Cada dispositivo que se conecta a la red se denomina “nodo”. Estos no pueden enviar mensajes directamente a otros nodos, ya que envían sus mensajes a la red del bus CAN, donde estarán disponibles para cualquier otro nodo al que vaya dirigido el mensaje. Como se describe cada nodo consta de los siguientes componentes:

- Unidad de control: se encarga de decidir la información a ser transmitida o recibida, así como también de la interpretación de cada uno de los mensajes.
- Controlador CAN: se comunica con el microprocesador de la unidad de control para acondicionar los datos y enviarlos al transceptor CAN, así mismo recibe los datos del transceptor, los acondiciona y envía al microprocesador de la unidad de control.
- Transceptor CAN: transmite y recibe los datos directamente del bus CAN. Se encarga de acondicionar la información para que sea utilizada por los controladores, situándola en los niveles de tensión adecuada.
- Resistencias de terminación: se conectan en los extremos del cable del bus, para evitar fenómenos de reflexión. Estas están conectadas en cada extremo, en paralelo con un valor de 120Ω, por lo cual si el sistema funciona correctamente la resistencia de la línea CAN debería ser de 60Ω.
- Cables del bus: consiste en dos cables trenzados para unir todas las unidades de control del sistema. Puede alcanzar una longitud de 500 m con una velocidad máxima de transmisión de 125 kbps, hasta 100 m con una velocidad de 500 kbps y 40 m a 1 Mbps. En todos estos casos se recomienda no exceder 50 cm entre el bus y el dispositivo conectado.

Para la transmisión de los datos, el controlador realiza una petición de inicio de transmisión, junto con todos los datos del mensaje. Cuando el bus está libre el mensaje es transmitido a todas las unidades de control, las cuales verifican el identificador y determinan si el mensaje es requerido, caso contrario lo ignoran. Este mensaje es una secuencia de bits con diferentes parámetros para identificarlos de inicio a fin y permitir la comunicación entre diferentes unidades. Los mensajes pueden estar en un formato base con un identificador de 10 bits o en un formato extendido con un identificador de 29 bits, lo cual se encuentra definido en la norma ISO 15765-4.⁽¹⁾ En la figura 10, se muestra la estructura que tienen los mensajes.

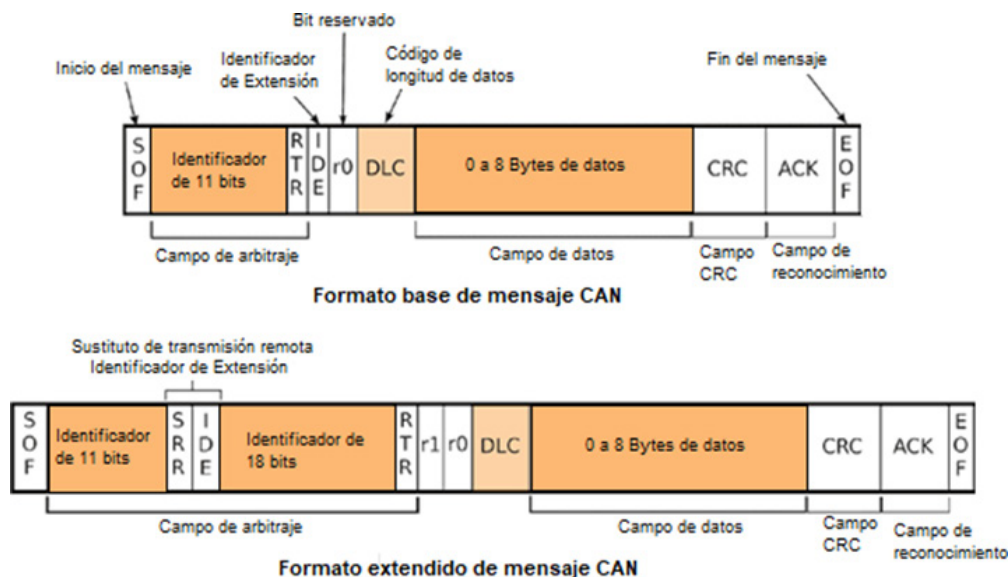


Figura 10. Estructura de mensajes del protocolo de comunicación CAN⁽¹⁾

Los campos de la estructura del mensaje especificados en la figura 11, se describen con mayor detalle en el Apéndice B, específicamente para el formato estándar de mensajes CAN. Mientras que, para el formato extendido se utilizan los mismos campos que en el formato estándar, pero se añade un campo denominado Solicitud Remota de Sustitución (SRR, Substitute Remote Request), que consiste en un bit recesivo ("1" lógico) para garantizar que una trama con un identificador de 11 bits, pueda acceder al bus al mismo tiempo que una trama remota con una extensión de ID de 29 bits. Y también se incluye un bit adicional (R1) reservado.

Al implementar esta red de comunicación en un vehículo, se puede tener un esquema de conexión como el que se presenta en la figura 11, en este caso se cuenta con dos buses CAN destinados a conectar los diversos módulos para el control del vehículo. A diferencia de un sistema convencional en donde la conexión punto a punto requiere de gran cantidad de cableado, en esta red se simplifica en gran medida la conexión de cada módulo.

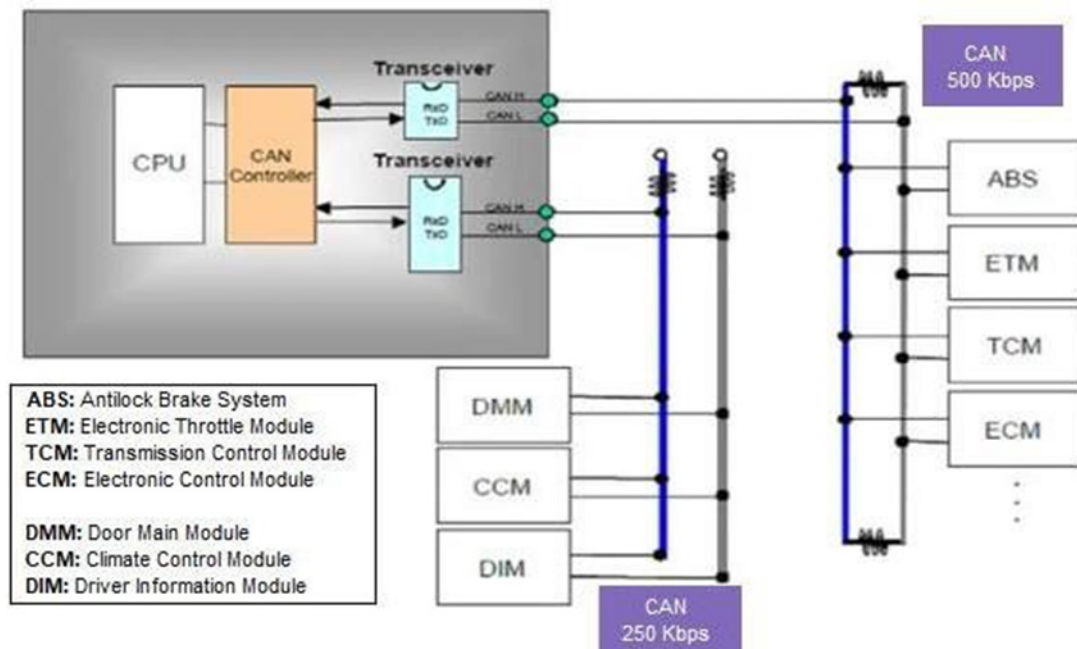


Figura 11. Esquema de conexión de un bus CAN en un vehículo

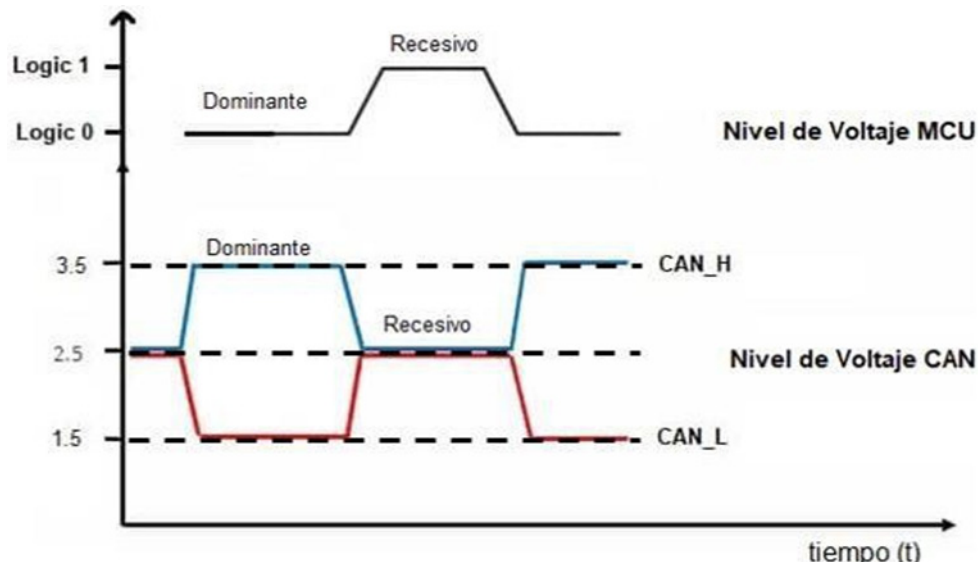


Figura 12. Niveles de voltaje del protocolo de comunicación CAN en un vehículo

Cabe destacar que al tener dos buses de comunicación CAN, los módulos principales del vehículo se conectan al bus CAN de alta velocidad (500 kbps), ya que se requiere una respuesta rápida de estos módulos porque son fundamentales para el correcto funcionamiento del vehículo. Tales como los módulos de control de transmisión (TCM), módulo del control electrónico (ECM), módulo de aceleración electrónica (ETM), entre otros. Mientras que otros módulos como el control de clima o puertas, tienen una prioridad menor, por lo cual se conectan al

bus CAN de menor velocidad (250 kbps).

Como se detalla para un bus CAN con una velocidad entre 40 kbps y 1 Mbps, los niveles de voltaje ideales son:

- Nivel de Voltaje Dominante: 1,5 V en CAN-L y 3,5 en CAN-H. El voltaje de CAN-H debe ser al menos 0,9 V más alto que CAN-L.
- Nivel de Voltaje Recesivo: 2,5 V en CAN-L y 2,5 en CAN-H. El voltaje de CAN-H no debe ser mayor al voltaje de CAN-L más 0,5 V.

En la figura 12, se ilustran los niveles de voltaje durante la transmisión de datos en el bus CAN y su relación con los niveles de voltaje interpretados por el microcontrolador del vehículo. Sin embargo, estos niveles de voltaje son solo referenciales ya que en la práctica pueden variar debido a varios factores como el estado del vehículo, la cantidad de unidades de control, entre otros.

CONCLUSIONES

El análisis del diseño e implementación de un sistema de enlace entre Alexa Auto y el computador del vehículo evidencia el impacto que las tecnologías digitales tienen en la industria automotriz contemporánea. La integración de asistentes de voz, en particular Alexa, con los estándares de diagnóstico como OBDII y el protocolo CAN, abre un horizonte de aplicaciones que trasciende el simple entretenimiento y se adentra en la optimización de la seguridad, la eficiencia y el mantenimiento vehicular. Este avance supone no solo una mejora en la experiencia del usuario, sino también un aporte a la reducción de riesgos asociados a la conducción, al disminuir las distracciones y brindar información en tiempo real de manera accesible y natural.

Los estudios revisados confirman que la adopción de asistentes de voz en los vehículos constituye una tendencia en constante expansión, en la medida en que los conductores valoran la posibilidad de interactuar con el sistema sin apartar la vista de la carretera ni las manos del volante. Asimismo, la incorporación de skills personalizadas y plataformas IoT refuerza la versatilidad de Alexa Auto, permitiendo que el asistente no solo responda a necesidades de confort, sino también a requerimientos críticos de monitoreo del estado del vehículo. Este enfoque sitúa al usuario en el centro, otorgándole un control más intuitivo y completo sobre su entorno de conducción.

De igual modo, el estudio del protocolo CAN y del estándar OBDII demuestra que la robustez y estandarización de estos sistemas hacen posible una comunicación confiable y ágil entre los módulos del vehículo, lo que garantiza diagnósticos más precisos y una detección temprana de fallas. Enlazar esta capacidad con un asistente de voz como Alexa implica avanzar hacia un ecosistema automotriz inteligente, donde la conectividad, la inteligencia artificial y la usabilidad convergen para crear soluciones preventivas y personalizadas.

En síntesis, la investigación evidencia que el desarrollo de un sistema de integración entre Alexa Auto y el computador del vehículo no es únicamente una innovación tecnológica, sino una propuesta estratégica para la movilidad del futuro. Su implementación contribuirá a optimizar la interacción entre conductor y vehículo, mejorar la seguridad vial, extender la vida útil de los automóviles y fomentar un uso más eficiente de los recursos. Con ello, se reafirma que la sinergia entre asistentes de voz, IoT y protocolos de diagnóstico no solo redefine el presente de la industria automotriz, sino que sienta las bases para una transición ordenada hacia la conducción autónoma y sostenible.

REFERENCIAS BIBLIOGRÁFICAS

1. Smith S. Digital voice assistants in use to triple to 8 billion by 2023. Juniper Research; 2019.
2. Calleja J. On the use of virtual assistant for smart cities parking service deployment. Santander: Universidad de Cantabria; 2021.
3. Caicedo K, Boderó L. Diseño e implementación de un sistema de conectividad para el monitoreo en tiempo real de los sistemas funcionales de un automóvil. Guayaquil: Universidad Politécnica Salesiana; 2021.
4. Apple. iOS - CarPlay. España: Apple; 2021.
5. Android Auto. Compatibilidad con Android Auto. 2017.
6. Amazon. ¿Qué es el Alexa Skills Kit? 2024.
7. Amazon. ¿Qué es una Skill de Alexa? 2024.
8. Amazon. Certify and Publish Your Skill. 2024.
9. Amazon. Design Your Skill. 2024.

10. Domínguez J. Desarrollo de una skill educativa sobre el asistente virtual Alexa: estudia conmigo. Elche: Universidad Miguel Hernández; 2022.
11. Amazon. Echo Auto - Pon Alexa en tu coche. 2024.
12. Muñoz C. Cómo crear tus skills de Alexa, aunque no sepas de programación. Tuexperto.com; 2022.
13. Voiceflow. Quick start guide. 2024.
14. Google Cloud. Cómo implementar un chatbot de IA con Dialogflow. 2020.
15. IONOS. Programación visual: la entrada más sencilla al mundo digital. 2020.
16. Edu J, Ferrer X, Such J, Suarez G. SkillVet: Automated Traceability Analysis of Amazon Alexa Skills. *IEEE Trans Depend Secure Comput.* 2021;20:161-75.
17. Tahsim A. Una descripción general de los conceptos básicos del diseño en Voiceflow. Voiceflow; 2023.
18. Tahsim A. Paso API: ¿Cómo hago llamadas API? ¿Cómo conecto mi Voiceflow a las API? Voiceflow; 2023.
19. Amazon. Lleva a Alexa en tu vehículo: Todo lo demás. 2024.
20. Amazon. Test and Debug Your Skill. 2024.
21. Blynk.io. Introduction - Blynk Documentation. 2024.
22. Villamar I. Estudio y análisis de los sistemas de diagnóstico en los automóviles modernos sistemas OBD. Cuenca: Universidad de Azuay; 2018.
23. ElectrOnline. MINI ELM327 V2.1 Bluetooth HH OBD OBDII avanzado calidad. 2024.
24. Falch M. OBD2 explained - a simple intro. 2023.
25. AutomotrizEscaner. Escáner Automotriz ELM327. 2024.
26. AV Electronics. Módulo WiFi ESP8266EX. 2024.
27. AV Electronics. Arduino Nano. 2024.
28. Electronilab. Módulo CAN bus MCP2515. 2024.
29. ENDADO. Sensor temperatura del refrigerante HELLA 89422-35010. 2024.
30. Fernández F. La sonda lambda o sensor de oxígeno. 2012.
31. Fernández R. Amazon Alexa Auto Tech: The ultimate guide to Alexa in your car. MUO. 2022.
32. Gonzales P. Tecnología HC. Los coches que mejor te hablan. 2022.
33. HELLA. Sensor de temperatura del refrigerante. 2024.
34. Litnevskiy S. Manual de usuario de Excel Mazda SkyActiv OBD-II calc (FORScan). 2021.
35. MACTRONICA. Convertidor DC-DC 12V a 5V 1A reductor. 2024.
36. Mazda CX-5 Service & Repair Manual: Controller Area Network (CAN) System. 2016.
37. Motores Auto. Sistema de inyección: partes, funcionamiento, tipos. 2023.
38. Naylamp Mechatronics. Módulo CAN MCP2515. 2023.

39. Opinautos. ¿Qué es el sensor de oxígeno (o sonda lambda) en un Hilux? 2024.
40. Ortiz A, Dávila R. Implementación de un asistente virtual para los estudiantes de pregrado de una universidad peruana. *Rev Conrado*. 2023;19(92):121-8.
41. Rudrawar K, Choudhar N, Meshram A. Voice assisted bots for automobile applications. In: *Proceedings of the 3rd International Conference on Advanced Technologies for Societal Applications*. Springer; 2021. p. 489-97.
42. STP. Sistema de combustible y aditivos de combustible. STP; 2023.
43. Toyota Venza. Toyota Venza: system diagram. Toyota Venza; 2016.
44. Transductor.net. Apertura eléctrica de puertas actuadores. 2023.
45. Varga A. CANDRIVE. GitHub; 2023.
46. Villén J. Simulador de la ECU de un vehículo con protocolo ISO 9141-2. Sevilla: Universidad de Sevilla; 2016.

FINANCIACIÓN

Los autores no recibieron financiación para el desarrollo de la presente investigación.

CONFLICTO DE INTERESES

Los autores declaran que no existe conflicto de intereses.

CONTRIBUCIÓN DE AUTORÍA

Conceptualización: Alexis Javier Villacis Ninasunta, Mariana Pinargote Basurto.

Curación de datos: Alexis Javier Villacis Ninasunta, Mariana Pinargote Basurto.

Análisis formal: Alexis Javier Villacis Ninasunta, Mariana Pinargote Basurto.

Redacción - borrador original: Alexis Javier Villacis Ninasunta, Mariana Pinargote Basurto.

Redacción - revisión y edición: Alexis Javier Villacis Ninasunta, Mariana Pinargote Basurto.