Facultad de Ciencia y Tecnología
Tecnología Superior en Electrónica Automotriz

Trabajo de Titulación:
Elaboración de un módulo para generación y obtención de señales CAN mediante microcontroladores y programación Arduino.

Trabajo previo a la obtención del título de Tecnólogo Superior en Electrónica Automotriz.

Autor:
Juan Fernando Velasco Vázquez

Director:
Ing. Cristian Jaramillo

Cuenca – Ecuador
2024

Dedicatoria

Este trabajo está dedicado a mi familia y amigos, cuyo apoyo incondicional ha sido fundamental en la culminación de este proyecto.

Agradecimientos

Agradezco profundamente a mi amigo, Ing. Diego Rojas, por su invaluable apoyo y asesoría durante este proceso. Asimismo, expreso mi gratitud a la Universidad del Azuay por brindarme las herramientas y el conocimiento necesarios para llevar a cabo este trabajo.

Resumen

El presente trabajo constituye un informe técnico sobre la elaboración de un módulo para la generación, obtención y visualización de señales CAN mediante microcontroladores compatibles con la programación Arduino. Se propone una herramienta económica y sencilla para el análisis de señales CAN, este módulo será capaz de simular varios componentes que constituyen una red CAN de un vehículo convencional. Se realiza una selección cuidadosa de componentes y se programa un código que permite la transmisión y recepción de señales CAN, con pruebas realizadas a través del monitor serial y un osciloscopio para asegurar la integridad de las señales.

Palabras clave: Protocolo CAN, microcontroladores, Arduino, red vehicular, señales CAN, análisis de fallas.

Abstract

This work presents a technical report on the development of a module for generating, obtaining, and visualizing CAN signals using microcontrollers compatible with Arduino programming. It proposes an economical and simple tool for CAN signal analysis, this module will be capable of simulating various components that constitute a CAN network of a conventional vehicle. A careful selection of components is made, and code is programmed to enable CAN signal transmission and reception, with tests conducted through the serial monitor and an oscilloscope to ensure signal integrity.

Keywords: CAN Protocol, microcontrollers, Arduino, vehicular network, CAN signals, fault analysis.

Índice de contenido

Índice de tablas

Índice de figuras

1 Introducción

Esta investigación propone la implementación de un módulo que genere, obtenga y visualice la señal de protocolo CAN mediante microcontroladores que soporten la programación de Arduino, proporcionando una herramienta simple y económica al análisis de este protocolo que se encuentra implementado en la mayoría de vehículos en la actualidad.

El análisis de fallas y de comportamiento de los dispositivos electrónicos incluidos en los vehículos se puede censar mediante el protocolo CAN, que desde su primera implementación en los años 80 y la consecuente aplicación como el estándar ISO 11898 (ISO, 2003) conocido como CAN 2.0 A (Bosch, 1991) que prácticamente todos los vehículos tienen en la actualidad, nos conlleva a utilizar dispositivos que obtengan la información que este protocolo nos brinda y así poder analizar los datos que generan los componentes que conforman la red CAN del vehículo.

En general, el objetivo de esta investigación es proporcionar una herramienta económica y fácil de desarrollar para el análisis y estudio de las señales CAN que existen en el vehículo, tomando este punto de partida para poder verificar que existen datos coherentes con los entregados en dicho protocolo. Con la verificación de una señal CAN coherente al estándar, podemos ahorrar tiempo y dinero corroborando que no existe daño en la red y así poder actuar directamente en otros desperfectos que podría presentar el vehículo.

2 Objetivos

2.1 Objetivo General

Elaborar un módulo para generar y obtener señales CAN mediante microcontroladores y programación Arduino.

2.2 Objetivos específicos

- Investigar los microcontroladores compatibles con Arduino disponibles en el mercado y seleccionar el más adecuado para el proyecto.
- Desarrollar el código necesario para la transmisión y recepción de señales CAN utilizando Arduino.
- Visualizar la señal CAN en una pantalla del módulo para verificar su coherencia. Analizar los datos adquiridos utilizando el monitor serial del Software IDE en un computador.

3 Procedimiento

3.1 Investigación y Selección de Componentes

Se llevó a cabo una investigación exhaustiva sobre microcontroladores compatibles con Arduino, enfocándose en las familias ESP32 de Espressif Systems y ATmega de Atmel. Se evaluaron sus características técnicas, disponibilidad en el mercado y relación costo-beneficio.

Tabla 1 Análisis comparativo de microcontroladores

Microcontrolador	Prestaciones	Costo (USD)	Compatibilidad
ATmega328P (Arduino Uno)	32 KB Flash, 2 KB RAM, 16 MHz	25	Alta
ATmega2560 (Arduino Mega)	256 KB Flash, 8 KB RAM, 16 MHz	40	Alta
AT91SAM3X8E (Arduino Due)	512 KB Flash, 96 KB RAM, 84 MHz	50	Media
ATmega328P (Arduino Nano genérico)	32 KB Flash, 2 KB RAM, 16 MHz	10	Alta

Tras el análisis, se optó por el Arduino Nano genérico con el microcontrolador ATmega328P (Inc., ATmega328/P Datasheet [Datasheet], 2014) debido a su equilibrio entre funcionalidad y costo, su tamaño compacto y su amplia compatibilidad con librerías y módulos de Arduino (Banzi, 2014).

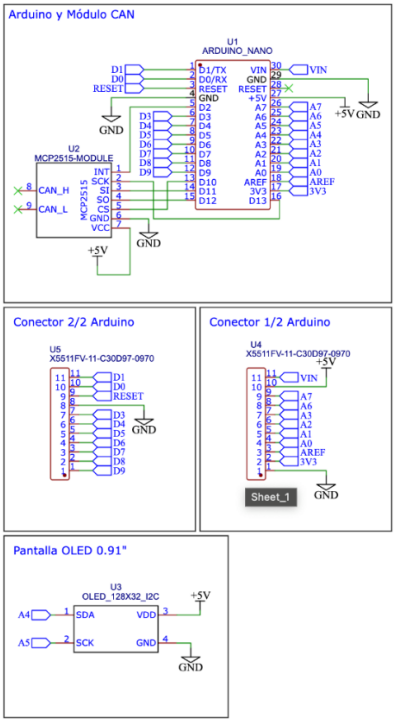
Para la comunicación CAN, se seleccionó el módulo MCP2515, que incluye el controlador CAN MCP2515 y el transceptor MCP2551, facilitando su integración con el Arduino Nano (Inc., MCP2515 Stand-Alone CAN Controller with SPI Interface [Datasheet]., 2016). Además, se incorporó una pantalla OLED de 0.91" con interfaz I2C para visualizar información en tiempo real, sin ocupar muchos pines (Industries, 2019).

4 Diseño del Hardware

4.1 Esquemático del Sistema

Se elaboró el diagrama esquemático del sistema que muestra la interconexión entre el Arduino Nano, el módulo MCP2515 y la pantalla OLED. El microcontrolador se comunica con el MCP2515 mediante SPI y con la pantalla OLED mediante I2C.

Figura 1 Esquemático del sistema con Arduino Nano, MCP2515 y pantalla OLED

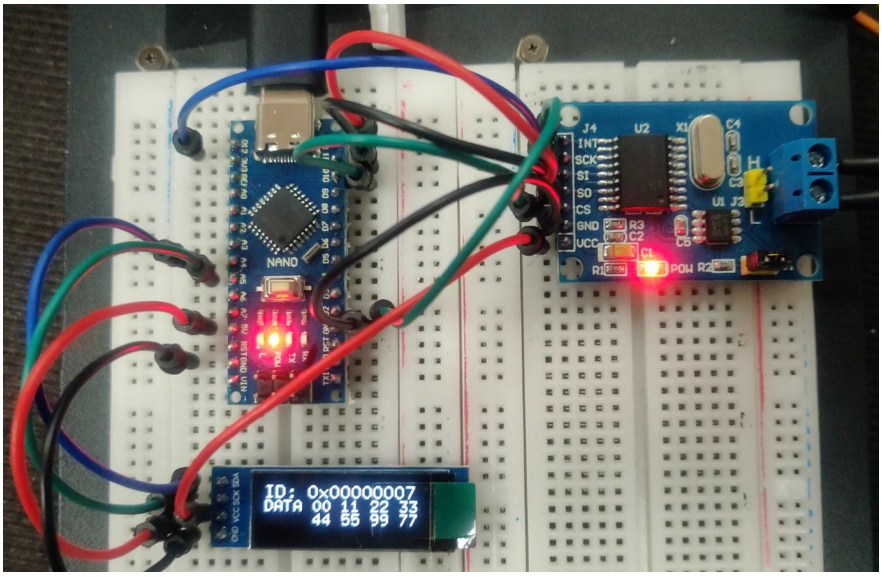


Fuente: El autor

4.2 Conexión del Módulo MCP2515 al Arduino Nano

Se realizó la conexión física en Protoboard entre el módulo CAN y el Arduino Nano, como se muestra en la siguiente figura.

Figura 2 Conexión en Protoboard entre el Arduino Nano y el módulo MCP2515



Fuente: El autor

La conexión entre el Arduino Nano y el MCP2515 se realiza a través de la interfaz SPI y se detalla a continuación:

Interfaz SPI:

MOSI (D11) conectado a SI del MCP2515.

MISO (D12) conectado a SO del MCP2515.

SCK (D13) conectado a SCK del MCP2515.

CS (D10) conectado a CS del MCP2515.

Interrupción:

INT (D2) conectado a INT del MCP2515.

Alimentación:

5V conectado a VCC del MCP2515.

GND conectado a GND del MCP2515.

Conexión de la Pantalla OLED al Arduino Nano

La pantalla OLED se conecta mediante I2C:

Interfaz I2C:

SDA (A4) conectado a SDA de la pantalla OLED.

SCL (A5) conectado a SCL de la pantalla OLED.

Alimentación:

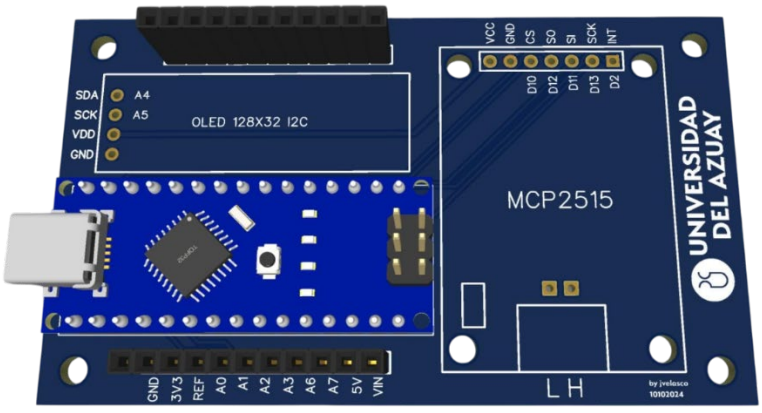
5V conectado a VCC de la pantalla OLED.

GND conectado a GND de la pantalla OLED.

4.3 Diseño del Circuito Impreso (PCB)

Se diseñó un PCB personalizado utilizando software EasyEDA para facilitar el montaje y reducir el tamaño del módulo. Se consideraron buenas prácticas de diseño para asegurar la integridad de las señales.

Figura 3 Diseño del PCB del módulo

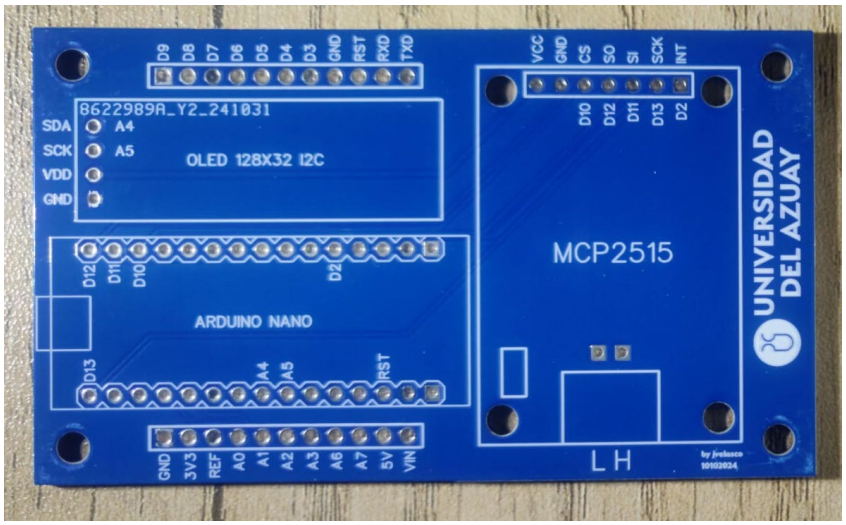


Fuente: El autor

4.4 Proceso de Fabricación del PCB

El circuito impreso fue fabricado por la empresa JLCPCB, ubicada en China, la cual brinda servicios profesionales de manufactura electrónica.

Figura 4 Fabricación del PCB



Fuente: El autor

4.5 Integración y Montaje

Se ensamblaron los componentes en el PCB siguiendo el diseño, asegurando conexiones y soldaduras correctas.

Pasos:

Montaje de componentes:

Soldadura del Arduino Nano, MCP2515 y pantalla OLED.

Verificación de conexiones:

Uso de multímetro para comprobar continuidad.

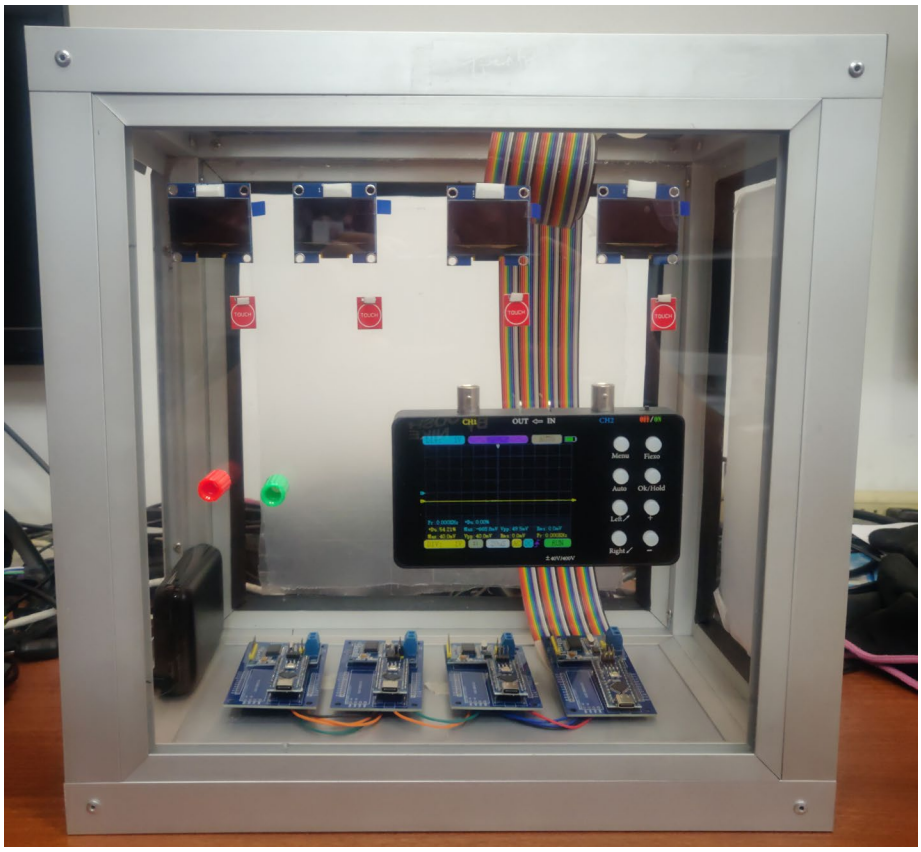
Alimentación:

Verificación de voltajes adecuados.

4.6 Fabricación del Módulo

El Módulo se construyó utilizando una estructura principal de perfiles de aluminio, elegidos por su resistencia y durabilidad. Los paneles y paredes se fabricaron con láminas de acrílico transparente de 4mm de espesor, lo que facilita la visualización de los componentes internos. La unión entre materiales se realizó mediante tornillería específica y elementos de fijación adecuados para garantizar la estabilidad del conjunto.

Figura 5 Módulo Completo



Fuente: El autor

5 Desarrollo del Software

El código se desarrolló en el IDE de Arduino, integrando las librerías MCP_CAN Library para el MCP2515 y Adafruit SSD1306 para la pantalla OLED (Patterson, 2015).

Funciones implementadas:

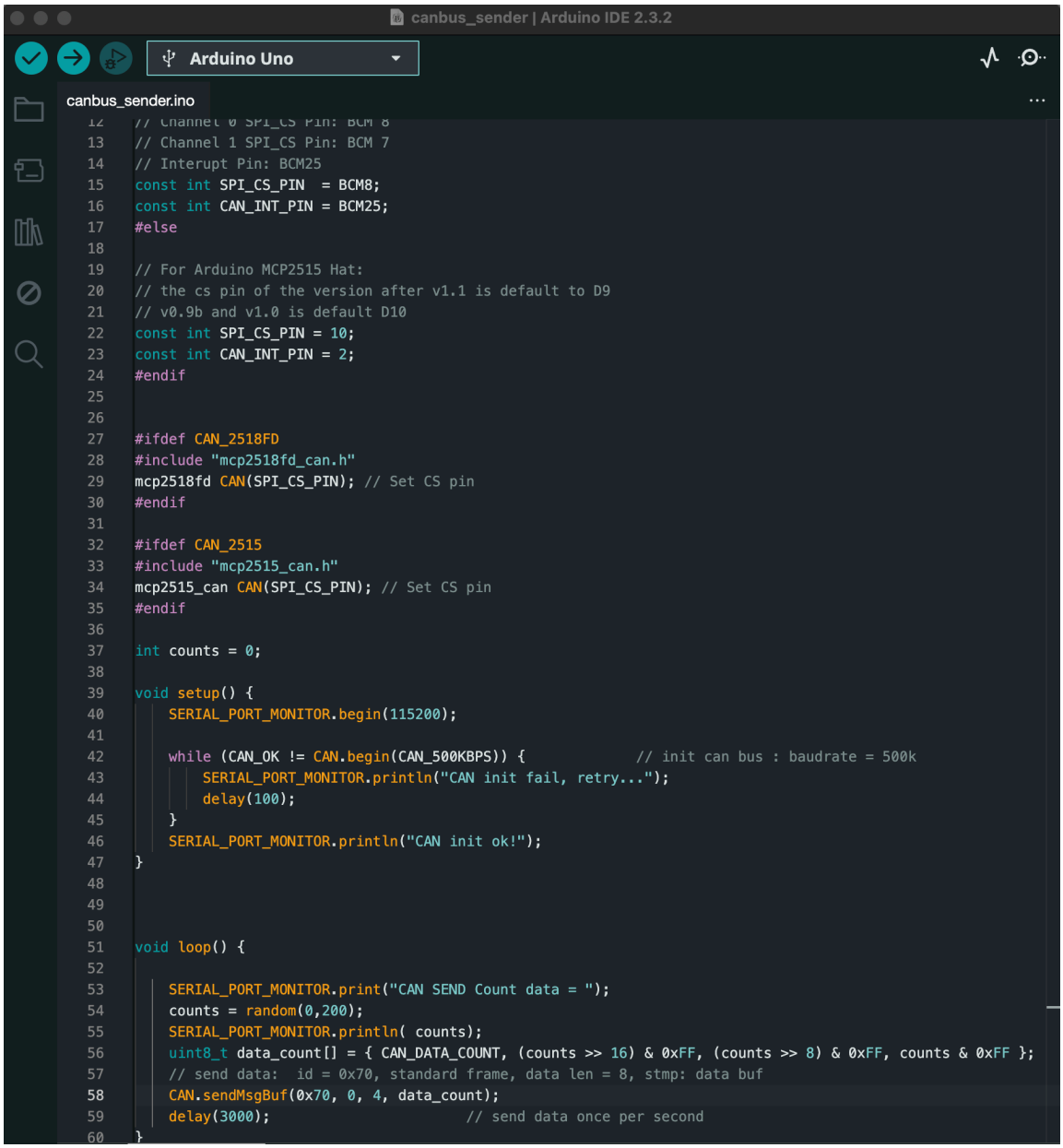
Inicialización del bus CAN a 500 kbps.

Transmisión y recepción de mensajes CAN.

Visualización en pantalla OLED de ID y datos recibidos.

Envío de datos al monitor serial para análisis detallado.

Figura 4. Fragmento de código principal:



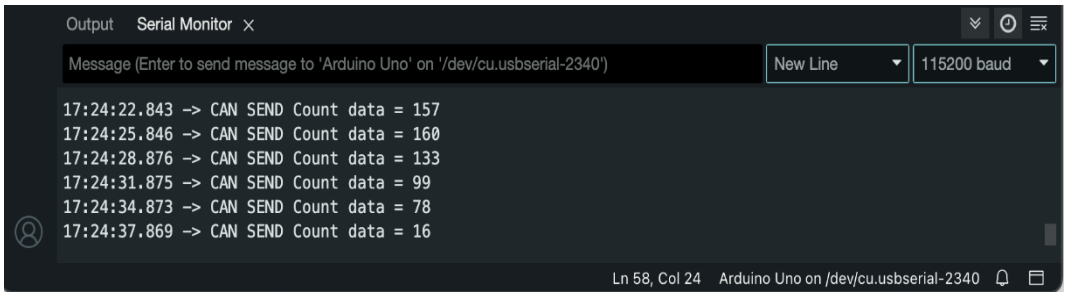
Fuente: El autor

6 Pruebas y Validación

6.1 Pruebas Iniciales

Se realizaron pruebas de transmisión y recepción entre dos módulos idénticos, visualizando datos en la pantalla OLED y monitor serial (Ali, 2015).

Figura 5. Monitor serial mostrando datos CAN

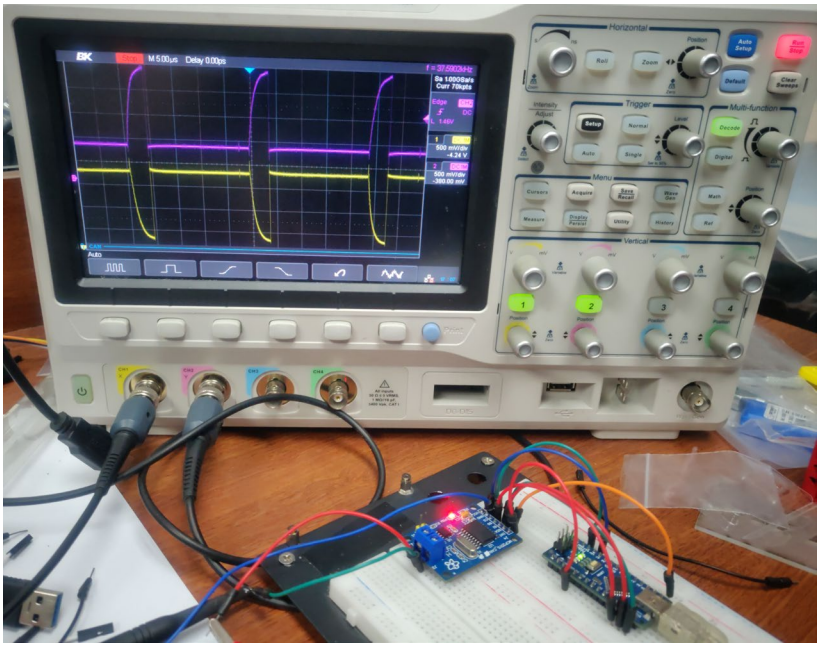


Fuente: El autor

7 Visualización en Osciloscopio

Se utilizó un osciloscopio para verificar la integridad de las señales CAN.

Figura 6. Señal CAN en osciloscopio



Fuente: El autor

8 Resultados

El módulo desarrollado es capaz de generar y recibir señales CAN de forma confiable, cumpliendo con los estándares del protocolo. La información se muestra correctamente en la pantalla OLED y monitor serial.

Ventajas:

Costo reducido: Componentes accesibles y económicos.

Portabilidad: Diseño compacto y ligero.

Flexibilidad: Código adaptable para futuras mejoras.

Tabla 2 Presupuesto del proyecto

Ítem	Cantidad	Costo Unitario (USD)	Costo Total (USD)
Arduino Nano (genérico)	5	10	50
Módulo MCP2515 CAN Bus	5	5	25
Pantalla OLED 0.91" I2C	5	8	40
Materiales para PCB	1	5	5
Cables y conectores	Varios	5	5
Componentes electrónicos varios	Varios	5	5
Total			**130**

9 Conclusiones

La implementación de un módulo para la generación y recepción de señales CAN utilizando el ATmega328P en un Arduino Nano, junto con el MCP2515 y la pantalla OLED I2C, ha demostrado ser una solución efectiva y económica. El proyecto cumple con los objetivos planteados, ofreciendo una herramienta útil para el diagnóstico y análisis de sistemas electrónicos vehiculares.

El diseño y fabricación del PCB mediante métodos accesibles permite replicar el módulo de forma sencilla, fomentando el aprendizaje práctico en electrónica. El módulo facilita la verificación y análisis de señales CAN, ahorrando tiempo y recursos en detección de fallas.

10 Referencias

Ali, M. &. (2015). Implementation of CAN Bus on Arduino [Tesis de Licenciatura, Universidad de Linköping]. Universidad de Linköping.

Banzi, M. &. (2014). *Getting Started with Arduino (3ra ed.).* Maker Media.

Bosch, R. (1991). *CAN Specification Version 2.0.* Stuttgart: Robert Bosch GmbH.

Inc., M. T. (2014). ATmega328/P Datasheet [Datasheet]. Microchip Technology Inc.

Inc., M. T. (2016). MCP2515 Stand-Alone CAN Controller with SPI Interface [Datasheet]. Microchip Technology Inc.

Industries, A. (2019). Monochrome OLED Breakouts [Manual de usuario]. Adafruit Industries.

ISO. (2003). ISO 11898-1:2003 Road vehicles — Controller area network (CAN). ISO.

Patterson, R. (2015). *Controller Area Network Prototyping with Arduino.* Packt Publishing.