International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 09 Sep. 2024, pp: 123-127 www.ijaem.net ISSN: 2395-5252

Applying Labview Software to Create Simulated Errors on a Semi-Experimental Model of Toyota Camry Engine for Automotive Technology Training

Nam Tran Anh[1] and Triet Ho Dac Hien[1]
[1]*LILAMA2 International Technology College, Vietnam*

Date of Submission: 01-09-2024 Date of Acceptance: 10-09-2024

ABSTRACT: In the realm of automotive technology education, the acquisition of hands-on experience in identifying, diagnosing, and resolving various engine faults plays a pivotal role in cultivating students' diagnostic and problem-solving capabilities. Despite its significance, the replication of real-world fault conditions within an educational setting poses substantial challenges, encompassing both the complexity of accurately simulating such conditions and the potential risks and time constraints associated with their creation. Addressing these concerns, this paper introduces an innovative methodology for the simulation of engine errors, employing LabVIEW software in conjunction with a semi-experimental model of a Toyota Camry engine. This proposed approach facilitates the generation of realistic and diverse fault scenarios, thereby providing students with the opportunity to refine their diagnostic skills within a controlled, risk-free environment. By leveraging the capabilities of LabVIEW software, this method not only simulates a wide array of engine malfunctions ranging from minor sensor discrepancies to significant performance-degrading errors but also allows for the precise manipulation and observation of these faults in real-time.

Keywords: Automotive technology education, LabVIEW software, simulated engine errors, semi-experimental model, Toyota Camry engine, fault diagnosis, problem-solving skills, practical experience

I. INTRODUCTION

Automotive technology education requires students to have practical experience in diagnosing and repairing various engine faults. Traditional teaching methods often rely on creating real faults in engines, which can be costly, time-consuming, and potentially dangerous. To address these challenges, we propose the use of LabVIEW software to simulate engine errors on a semi-experimental model of a Toyota Camry engine. This approach offers a cost-effective and safe alternative to traditional methods, allowing students to gain valuable hands-on experience in a controlled environment.

The 3MZ-FE engine, an evolution within Toyota's MZ series, exhibits a displacement of 3.3 liters (3,310 cc) with a bore and stroke configuration of 92 mm × 83 mm (3.62 in × 3.27 in). This engine variant generates an output of 225 horsepower (168 kW; 228 PS) and 240 lb·ft (325 N·m) of torque when installed in the Toyota Camry, and slightly higher figures of 230 horsepower (172 kW; 233 PS) and 242 lb·ft (328 N·m) of torque in both the Toyota Sienna and Highlander models. Notably, the 3MZ-FE incorporates advanced technological features such as Variable Valve Timing with intelligence (VVT-i), an Electronic Throttle Control System with intelligence (ETCS-i), a PA6 plastic intake manifold, and an enlarged throttle body diameter compared to its predecessor, the 1MZ-FE.

A significant enhancement in the 3MZ-FE engine is the adoption of a new flat-type knock sensor, marking a departure from the resonator-type knock sensors utilized in prior MZ series engines. This advancement addresses the earlier models' challenges with knock detection sensitivity or inadequate knock control, which could result in a potential reduction of up to 20 horsepower (15 kW) due to suboptimal ignition timing when operating on fuel with an octane rating lower than 91. The flat-type knock sensor is engineered with a novel design that is capable of detecting a broader range of frequencies, thus supplying the Engine Control Unit (ECU) with more precise data for optimal engine performance. This sensor is mounted to the engine block on each bank via a central bolt, ensuring secure attachment and reliable operation.[1].

Fig 1. 3MZ-FE in 2004 Lexus ES330

This engine was effectively utilized in the design and construction of a semi-experimental model, a particularly crucial tool in enhancing the quality of teaching and learning in the field of automotive engineering. Utilizing LabVIEW software[2][3], one of the most powerful tools for digital simulation and control processes, this semi-experimental model not only allows for the simulation of actual faults but also facilitates an interactive and efficient experience for both students and instructors to explore, analyze, and devise solutions to various technical issues. By directly connecting to the control and monitoring system through LabVIEW, this model enables the simulation of engine operations under different conditions. This contributes to a deeper understanding of engine mechanics and the methodologies for detecting and addressing technical malfunctions. The integration of the semi-experimental model with LabVIEW software opens up new pedagogical methods, making learning more dynamic and allowing students to gain in-depth knowledge through practical exercises, thereby enhancing their professional skills and knowledge in automotive engineering. Furthermore, the creation of simulated faults through this semi-experimental model also makes the teaching and learning process more flexible. Instructors can easily set up various fault scenarios, ranging from simple to complex, providing students with opportunities to be challenged and apply their theoretical knowledge in real-world contexts. This approach fosters critical thinking, creativity, and problem-solving abilities in students.

II. DESIGN OF SEMI-EXPERIMENTAL MODEL TO CREATE CAR ENGINE FAULTS

1.1. Block diagram of the system

The schematic architecture of the system under discussion is delineated through the following components:

- A 12-volt accumulator block (Accu) that serves as the primary power source for the entire system. This component is crucial for ensuring a stable and continuous supply of electrical energy, facilitating the seamless operation of the system's components.
- An interface block that facilitates communication from the automobile engine to the computer. This interface is realized through the deployment of an Arduino Mega 2560 board coupled with a relay control unit. The Arduino Mega 2560, renowned for its robust performance and versatility, acts as a critical intermediary, translating the analog signals from the engine into digital data that can be processed by the computer. The relay control unit further enhances this interfacing capability, allowing for precise control over the engine's operational parameters.
- A computer that has been pre-installed with LabVIEW software. This software stands at the forefront of system control and data acquisition, offering a graphical programming environment that is both intuitive and powerful. LabVIEW's capabilities in simulating engine faults and facilitating real-time data analysis and control are unparalleled, making it an indispensable tool in the educational framework of automotive engineering.

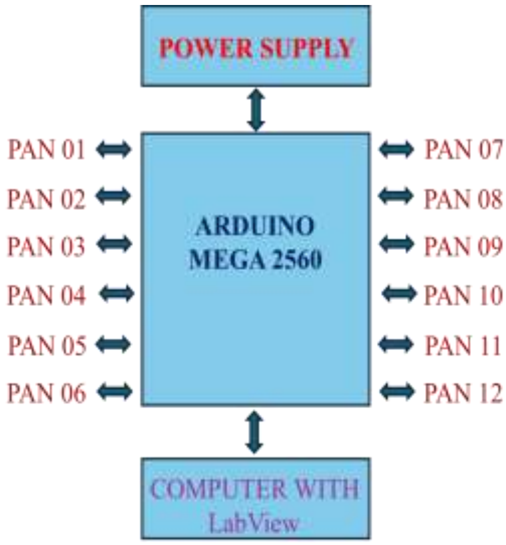

Fig 2. Block diagram of interface Arduino board and LabVIEW software

According to the instructional design, the instructor will manipulate the LabVIEW software to induce specific faults within the automobile engine. This deliberate introduction of faults serves a dual purpose: firstly, it provides a practical, hands-on learning experience for the students, allowing them to engage directly with the complexities and challenges of automotive diagnostics and repair. Secondly, it cultivates a deeper understanding of the engine's operational dynamics, as students are required to employ critical thinking and problem-solving skills to identify and rectify the faults generated by the instructor.

1.2. Arduino Mega 2560

The Arduino Mega 2560 Rev3, an authentic product made in Italy, represents an advanced iteration of the Arduino Uno, offering an expanded array of communication pins, peripherals, and memory capacity. This board is second only to the Arduino Uno in terms of popularity and is compatible with the majority of Arduino shields available today, facilitating a wide range of applications [4].

The genuine Arduino Mega 2560 Rev3 is recognized for its exceptional component quality, craftsmanship, and durability. It comes with a warranty and after-sales service that adhere to Arduino's high standards. To ensure authenticity, customers should seek out the official packaging, stickers, and anti-counterfeit labels provided by Arduino.cc. These features include clear manufacturing details on both the board and its packaging, helping to distinguish the genuine product from lower-quality counterfeit alternatives available in the market.

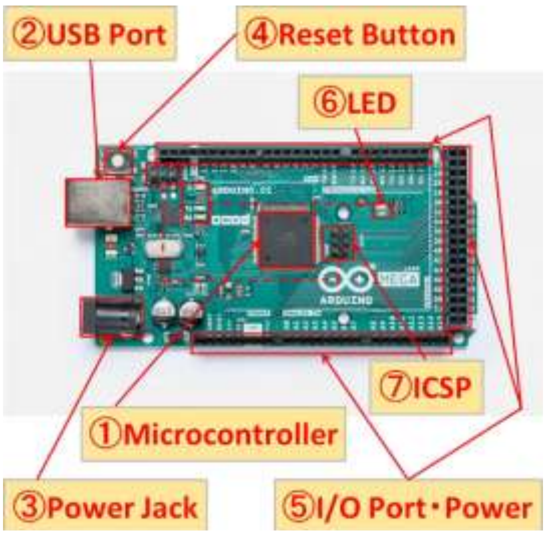

Fig 3. Arduino Mega 2650

Technical Specifications:
- Microcontroller: AT Mega2560
- Operating Voltage: 5V DC
- Recommended Input Voltage: 7~12VDC
- Input Voltage Limit: 6~20VDC
- Digital I/O Pins: 54 (15 provide PWM output)
- Analog Input Pins: 16
- DC Current per I/O Pin: 20 mA
- DC Current for 3.3V Pin: 50 mA
- Flash Memory: 256 KB (8 KB used by bootloader)
- SRAM: 8 KB
- EEPROM: 4 KB
- Clock Speed: 16 MHz
- Built-in LED: Pin 13
- Dimensions: 101.52 mm x 53.3 mm
- Weight: 37 g

2.3. LabView Software

The Laboratory Virtual Instrument Engineering Workbench (LabVIEW) constitutes a sophisticated platform for graphical system design and development, curated and disseminated by National Instruments. This platform is grounded in a programming paradigm that employs a visual language for programming, facilitating widespread application in domains such as data acquisition, instrument control, and industrial automation. Through its comprehensive suite of tools, LabVIEW empowers users to design and implement intricate test and measurement systems with enhanced efficiency.

The programming language integral to LabVIEW, known as "G" (distinct from G-code), is a dataflow language that was innovatively developed by National Instruments. This language's dataflow orientation allows for an intuitive representation of computational processes, thereby streamlining the development of applications within LabVIEW. The platform's compatibility spans across various operating systems, including macOS, several Unix and Linux distributions, and Microsoft Windows, underscoring its versatility and broad applicability.

Recent advancements in LabVIEW's evolution are marked by the releases of LabVIEW 2023 Q1 in April 2023 and LabVIEW NXG 5.1 in January 2021. In a significant move towards fostering a community of developers and enthusiasts, National Instruments introduced the LabVIEW and LabVIEW NXG Community editions on April 28, 2020, making them freely available for non-commercial use. This initiative not only democratizes access to high-level virtual instrumentation development tools but also encourages innovation and collaborative development among users worldwide [5].

International Journal of Advances in Engineering and Management (IJAEM)
Volume 6, Issue 09 Sep. 2024, pp: 123-127 www.ijaem.net ISSN: 2395-5252

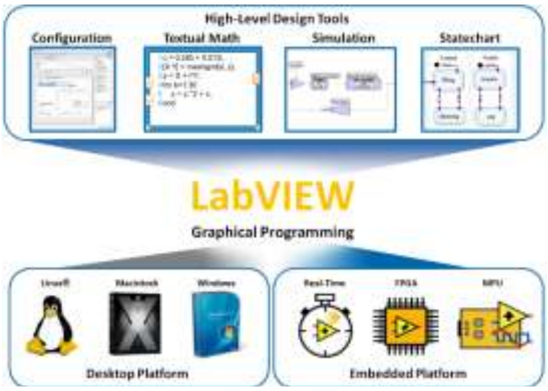

Fig 4. LabVIEW is a graphical programming system from National Instruments (NI)

III. SEMI-EXPERIMENTAL MODEL TO CREATE CAR ENGINE FAULTS

Within the context of automotive technology education, the utilization of LabVIEW software in conjunction with a semi-experimental model of a Toyota Camry engine represents a novel approach towards simulating engine faults. This methodology not only facilitates an immersive learning experience but also mitigates the risks and limitations associated with traditional hands-on practices. The following sections elaborate on the systematic procedure for deploying this innovative teaching tool.

Step-by-Step Operational Guide

Step 1: Installation and preparation

Ensure that the LabVIEW software is properly installed on the computer. Establish a connection between the Arduino Mega 2560 and the computer using a USB cable. Install the necessary drivers to facilitate the computer's recognition of the Arduino Mega 2560.

Step 2: Interface Design on LabVIEW

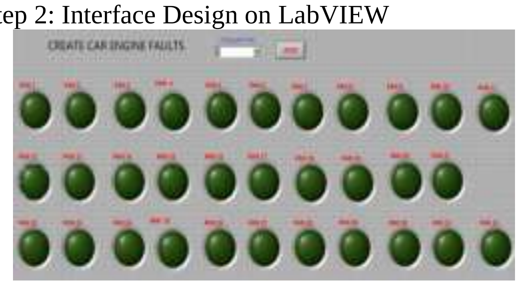
Fig 5. LabVIEWsoftware interface generate error codes

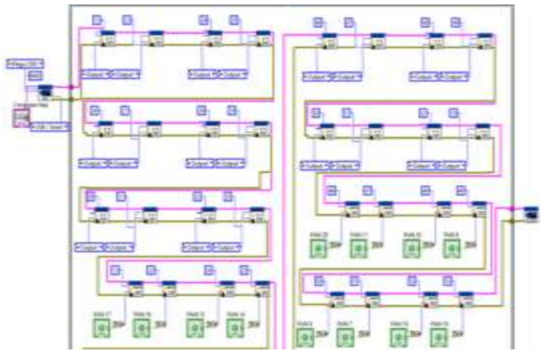
Fig 6. Graphic programming interface

Launch LabVIEW and initiate a new project. In the front panel environment, design a user interface (GUI) incorporating control buttons and data visualization graphs. In the block diagram environment, construct the control logic by interconnecting functional blocks.

Step 3: Programming and Connecting Arduino

Employ the LabVIEW Interface for Arduino (LIFA) or LINX to program and upload code to the Arduino. Utilize corresponding blocks in LabVIEW to transmit and receive data to/from Arduino.

Step 4: Simulating Engine Faults

Configure engine fault scenarios within LabVIEW, including conditions such as overheating, oil pressure loss, or sensor malfunctions. Use the developed interface to activate and manage these simulated faults on the engine model.

Step 5: Analysis and Resolution


Fig 7. Real model connected to computer using LabView software

Leverage the LabVIEW interface to observe and analyze data from the engine model, identifying faults. Proceed to resolve these faults based on the analysis, employing acquired knowledge and resources from LabVIEW.

IV. CONCLUSION AND FUTURE DIRECTIONS

The adoption of LabVIEW software in tandem with a semi-experimental Toyota Camry engine model has heralded a groundbreaking methodology in automotive technology education. This approach not only minimizes costs and risks but also enhances educational effectiveness by enabling students to actively engage in fault diagnosis and resolution in a controlled setting. By fostering an interactive and dynamic learning environment, the integration of LabVIEW and the semi-experimental engine model has proven to be a formidable educational tool, instrumental in the development of future automotive engineers. This methodology empowers students to cultivate problem-solving skills, critical thinking, and practical competencies within a manageable framework, thereby significantly contributing to their professional readiness and technical proficiency in the automotive sector.

REFERENCES

[1]. https://en.wikipedia.org/wiki/Toyota_MZ_engine#1MZ-FE

[2]. Ju, C., Zhang, G. E., & Su, R. (2023, March). Engine Speed Measurement and Control System Design Based on LabVIEW. In The International Conference on Artificial Intelligence and Logistics Engineering (pp. 79-89). Cham: Springer Nature Switzerland.

[3]. Vikhe, P., Punjabi, N., & Kadu, C. (2014). Real time DC motor speed control using PID controller in LabVIEW. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 3(9), 12162-12167.

[4]. https://hshop.vn/arduino-mega-2560-chinh-hang-original-made-in-italy

[5]. https://www.ni.com/en/shop/labview.html?srsltid=AfmBOopJfEhmClHZbpce7P_pLBaTfyTIAWP8jV6RmyjL_3SXd-sjYFwC