

SOFTWARE-IN-LOOP-SIMULATION FOR INSTRUMENT CLUSTER

1ST P.SAMPATH KUMAR
Assistant Professor
Department of EEE
Bapatla Engineering College
Bapatla,India.

2nd M.NISSELA MARGARATE
Department of EEE
Bapatla Engineering College
Bapatla,India.

3rd M.V.PRAVEEN KUMAR
Department of EEE
Bapatla Engineering College
Bapatla,India.

4th S.PRAVEEN KUMAR
Department of EEE
Bapatla Engineering College
Bapatla,India.

Abstract - This paper contributes to enhancing time-to-market and ensuring the reliability and compliance of instrument clusters with safety standards. By utilizing Software-in-the-Loop (SIL) simulation, it streamlines development processes and fosters innovation in automotive design. As automotive technology rapidly evolves, SIL simulation becomes pivotal for efficient development and validation. Addressing challenges such as real-time data processing and integration with vehicle networks, SIL simulation reduces costs, mitigates risks, and improves software-intensive systems' quality. Its early validation and iterative development capabilities make it essential for engineers worldwide. This approach leverages communication with CAN and integration with ECUs to further enhance instrument cluster functionality.

Keywords - Instrument clusters, Software-in-the-Loop simulation, Reliability, Safety standards, Automotive design, CAN bus, Electronic Control Units (ECUs), Quality improvement.

I. INTRODUCTION

[6] An Instrument Cluster is an indispensable component in vehicles, providing drivers with a comprehensive array of information ranging from critical alerts to routine data. Key points regarding its functionality and importance include:

Display Interface : The cluster utilizes an LCD screen to ensure clear visibility of information sourced from various vehicle sensors.

Central Control : The Electronic Control Unit (ECU) or central control unit plays a pivotal role in receiving and prioritizing data from sensors, ensuring that the most relevant information is promptly relayed to the driver.

Protocol Facilitation: [1] The prioritization process is streamlined by the Controller Area Network (CAN) protocol, which employs unique identifiers for efficient data transmission.

Efficient Data Relay: Through the CAN Bus, data from diverse sensors is efficiently transmitted to the central control unit, enabling the display of vital metrics such as speed, RPM, and fuel level.

Component Requirements: Each node within the network necessitates specific components like

microcontrollers, CAN controllers, and CAN transmitters to facilitate seamless communication.

Robustness of CAN Protocol: Notably, the CAN protocol demonstrates robustness, enabling easy error detection and resolution, and offers high-speed, flexible message-based communication.

Enhanced Driver Experience: This approach [8] ensures that drivers receive accurate and timely information on the instrument cluster, thereby enhancing overall driving experience and safety. The formatter will need to create these components, incorporating the applicable criteria that follow.

Software-in-the-loop(SIL)simulation serves as a valuable method in software development and testing, particularly within the realm of embedded systems and control engineering. SIL simulation involves testing the software component of a system independently from the hardware. Initially, engineers develop the software using high-level programming languages like C, C++, or domain-specific languages. Following this, a simulation environment is created on a computer to mimic the behavior of the eventual hardware setup. This environment can simulate inputs from sensors, outputs to actuators, and overall system behavior.

Subsequently, the developed software is integrated into the simulation environment, allowing it to run within this virtual setting instead of on the actual hardware. Engineers can then conduct thorough testing by providing various inputs and observing how the software reacts, mirroring real-world scenarios. SIL simulation offers numerous advantages, including early testing capabilities, cost-effectiveness, and the ability to repeat and automate tests for consistent results.

However, SIL simulation also comes with limitations. It may not accurately capture all aspects of the real system's behavior, particularly timing-related issues or interactions with physical components that are challenging to model. Therefore, SIL testing is often complemented with other testing methods such as hardware-in-the-loop (HIL) simulation and real-world testing to ensure comprehensive software validation.

II. PROCESS FLOW

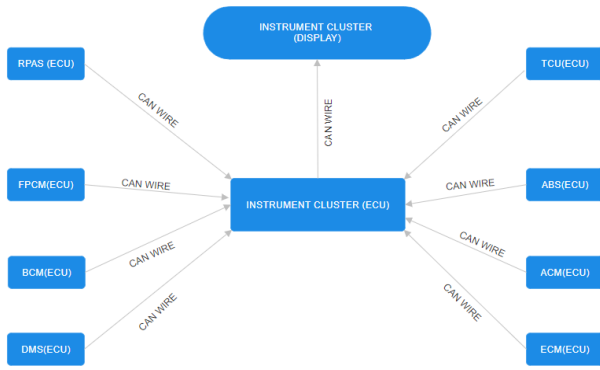


Fig 1.Communication between ECU's and Cluster

Utilizing Fig 1. the central role is assumed by the Instrument Cluster Electronic Control Unit (IC ECU). It gathers data from multiple Electronic Control Units (ECUs) including RPAS, BCM, ECM, TCU, ABS, DMS, ACM, and FPCM. Following this, the IC ECU transmits relevant information to the instrument cluster display, which then presents vital data to the driver for safety reasons.

A. ABBREVIATIONS :

ECU:Electronic Control Unit , IC:Instrument Cluter,
TCU:Transmission Control Unit , ABS:Antilock
Braking Sysem , ACM:Airbag Control Module ,
ECM:Engine Control Module , RPAS:Reverse
Parking Assist System , FPCM:Fuel Pump Control
Module , BCM:Body Control Module , DMS:Driver
Monitoring System

III. SYSTEM DESIGN

The Instrument Cluster boasts dimensions of 1200cm x 800cm and is intricately designed within the Panel window of the TSMaster Application.[2] Its construction involves the utilization of various properties such as Progress Bars, Text Boxes, Images, LEDs, Containers, and Gauges. Each LED incorporated within the cluster is uniquely customized, associating it with specific images and signals sourced from the CAN Database. This customization enables the dynamic representation of essential vehicle parameters like speed, fuel level, and warning indicators[3]. The integration of these diverse properties results in a comprehensive interface tailored for monitoring and controlling critical aspects of the vehicle's operation. Users are provided with a visually intuitive dashboard, significantly enhancing the overall driving experience and safety[7]. Through the TSMaster Application, the Instrument Cluster facilitates real-time data visualization, empowering drivers to make informed decisions while on the road. Fig 2.

A. SOFTWARES USED:

1. **TSMaster** : It is a versatile virtual instrument software designed to facilitate seamless connectivity, configuration, and management of TOSUN's hardware tools. Its functionalities span automotive bus monitoring, simulation, diagnostics, calibration, BootLoader, I/O control, measurement testing, and EOL. Additionally, TSMaster is equipped to handle

multiple protocols including LIN, CAN, CAN FD, FlexRay, and Ethernet, ensuring robust support for various applications.

2. **Arduino IDE** : It is a software development environment used for writing code for Arduino microcontroller boards. It provides a simple interface for writing, compiling, and uploading code to Arduino devices.

B. HARDWARES USED:

1. **In-Vehicle Test (IVN) Beds** : It feature a comprehensive array of components including digital switches, push buttons, analog switches, an ultrasonic sensor, an Arduino Mega 2560 Board linked to pin number 53, twelve LEDs, an LCD display, a buzzer, supplementary ports for sensor integration, a DC motor, and a CAN cable connector.
2. **TOSUN Interface** : The TC1011, a single-channel CAN (FD) bus interface device introduced by TOSUN, supports CANFD bus rates of up to 8M bps[4]. Featuring a high-speed USB 2.0 interface for PC connectivity, its drive-free design on Windows systems ensures excellent system compatibility.

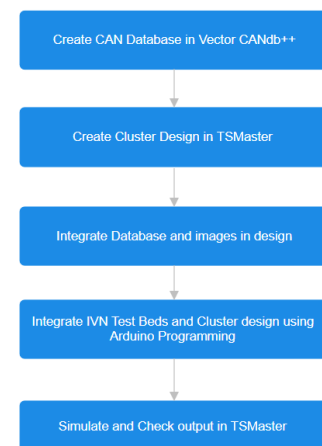


Fig 2.System Design

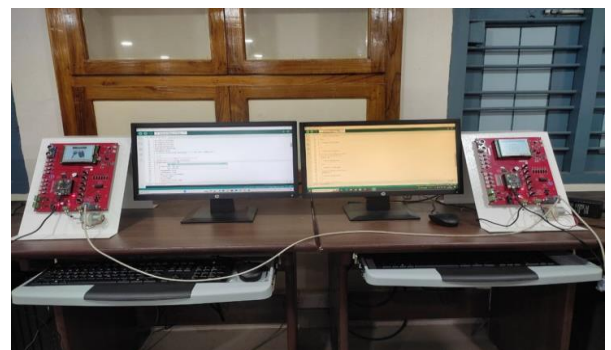


Fig 3. Connecting Two IVN Test Beds using CAN Cable

Fig 3. In this setup, we implemented two Arduino write codes to establish efficient communication between two IVN test beds and prevent message overflow into the MCP2515 board. Utilizing a CAN cable with two ends [5], one featuring a female port, we facilitated external hardware connections, including TOSUN hardware. This ensured seamless

communication between the systems, optimizing their performance and reliability. One interface is equipped with 5 signals while the other interface has 3 signals, with the capability to add more signals based on the creation of panels with different signals. Additionally, the IVN test bed is equipped with 8 analog pins, of which four were used for our panel purpose, along with 6 digital pins and 6 push button switches, enabling control of the panel in the TS Master tool.

digital pins, we were able to effectively simulate the desired outputs in the TS Master tool.



Fig 4. Instrument Cluster Design before simulation

Fig 4. This panel shows the design of the Instrument Cluster with integration of all signals, each assigned specific addresses obtained from the database. Observable are various types of gauges for speed, RPM, and throttle position, alongside progress bars for fuel composition. Additionally, LEDs with different colors such as red, green, and yellow indicate various signals. However, during the off condition of the simulation, the design does not illustrate movement in gauges or blinking in LEDs.

A. TRACE WINDOW

Absolute Time	Counter	Chn	Identifier	FFS	Message Name	Type	Dir	DL	DL C	Data Len.	BRS	ES2	RS	RS1	RS2	RS3	RS4	RS5	RS6	RS7	RS8	RS9	RS10	RS11	RS12	RS13	RS14	RS15
47.35426	4823	...	428	...	transmission_rpm...	Data	Tx	5	5	...	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
47.35426	4823	...	200	...	km/hr																							
47.35426	4823	...	0	...	rpm																							
47.35426	4823	...	75	...	C																							
47.35426	4823	...	2	...																								
47.35426	4823	...	3	...																								
47.35426	4823	...	4	...																								
47.35426	4823	...	5	...																								
47.35426	4823	...	6	...																								
47.35426	4823	...	7	...																								
47.35426	4823	...	8	...																								
47.35426	4823	...	9	...																								
47.35426	4823	...	10	...																								
47.35426	4823	...	11	...																								
47.35426	4823	...	12	...																								
47.35426	4823	...	13	...																								
47.35426	4823	...	14	...																								
47.35426	4823	...	15	...																								
47.35426	4823	...	16	...																								
47.35426	4823	...	17	...																								
47.35426	4823	...	18	...																								
47.35426	4823	...	19	...																								
47.35426	4823	...	20	...																								
47.35426	4823	...	21	...																								
47.35426	4823	...	22	...																								
47.35426	4823	...	23	...																								
47.35426	4823	...	24	...																								
47.35426	4823	...	25	...																								
47.35426	4823	...	26	...																								
47.35426	4823	...	27	...																								
47.35426	4823	...	28	...																								
47.35426	4823	...	29	...																								
47.35426	4823	...	30	...																								

Fig 6. Trace Window

The Trace window within the TS Master tool provides visibility into various signal types, their values dynamically updated in response to changes in digital and analog pins within the IVN test bed kit. Utilizing Tosun hardware, these signals are transmitted through the CAN wire to the TS Master software, facilitating comprehensive monitoring and analysis of automotive system behavior. Within the Trace window, essential details such as the absolute time of signal passage, identifier, frames per second, message name, and transmission direction (whether transmitting or receiving) are displayed. Additionally, the Trace window indicates the data length code, offering a comprehensive overview of signal transmission and reception.

B. GRAPHICAL WINDOW

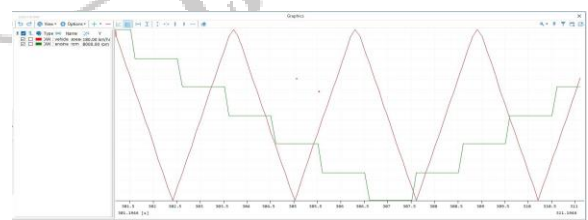


Fig 7. Graphical Display

Fig 7. The graphics window displays engine RPM in green and vehicle speed in red. By adjusting analog pins within the IVN test bed kit, these values can be varied to meet specific testing requirements in automotive applications, offering flexibility for customized displays.

IV. RESULTS



Fig 5. Instrument Cluster design after simulation

Fig 5. This output design shows the simulation panel using the transmit window, but the actual configuration was done with the IVN test bed. By utilizing various analog and digital pins, we enabled specific signals within the IVN test bed. This allowed us to observe the output in the simulation panel accurately. Additionally, we were able to activate LEDs and gauges through the transmit window in the TS Master tool. However, one drawback of this method is the inability to precisely control each signal as required. Despite this limitation, the IVN test bed provided a practical solution for configuring and observing specific signals in the simulation panel. Through careful selection and utilization of analog and

C. NUMERICAL DISPLAY



Fig 8. Numerical Display

The Numerical window offers comprehensive numeric representation of analog and digital pin fluctuations in the IVN test bed kit, providing real-time insight into signal variations during automotive testing. Detailed information, including signal type and timestamp, enhances contextual understanding. A bar graph within the window offers visual representation of signal trends, aiding in quick comprehension. Overall, this window enhances monitoring and analytical capabilities, contributing to efficient automotive testing processes.

V. CONCLUSION

The utilization of software-in-the-loop simulation for instrument cluster design represents a strategic approach to enhance development efficiency and product quality. Through this methodology, our project has demonstrated significant advantages including cost savings, accelerated development cycles, and improved accuracy in identifying potential issues. By leveraging virtual environments to simulate software behavior, we've achieved a more robust and refined Instrument Cluster design, ultimately leading to enhanced user satisfaction and streamlined production processes. This project underscores the importance of incorporating simulation techniques early in the design phase

to ensure successful outcomes in automotive instrumentation development.

VI. REFERENCES

- [1]. Surekha P. Gaikwad and Atmeshkumar S. Patel, "Automotive control system using CAN protocol", in *International Journal of Advanced Research, Ideas and Innovations in Technology*, 2019. ISSN: 2454-132X.
- [2]. G. Vora and P. Gundewar, "Survey on Designing of Electric Vehicle Instrument Cluster," *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, 2018, pp. 765-771.
- [3]. A. A. Salunkhe, Pravin P Kamble and Rohit Jadhav, "Design And Implementation of CAN bus Protocol for Monitoring Vehicle Parameters", in *IEEE International Conference On Recent Trends In Electronics Information Communication Technology*, May 20-21, 2016, India, ISBN: 978-1-5090-0774-5/16.
- [4]. S. N Chikhale, "Automobile Design and Implementation of CAN bus Protocol- A Review", in *IJRDO - Journal of Electrical and Electronics Engineering*, Volume-4/Issue 1/January 2018, ISSN: 2456-6055.
- [5]. Guilherme Marcon Zago and Edison Pignaton deFreitas, "A Quantitative Performance Study on CAN and CAN FD Vehicular Networks" in *IEEE Transactions On Industrial Electronics*, Vol-65, No.-5, May 2018, ISSN: 0278-0046.
- [6]. L. A. Perișoară, D. L. Săcălean and A. Vasile, "Instrument clusters for monitoring electric vehicles," *2017 IEEE 23rd International Symposium for Design And Technology in Electronic Packaging (SIITME)*, Constanta, 2017, pp. 379-382.
- [7]. Y. Tao, Y. Yue and P. Craig, "A hybrid approach to detection and recognition of dashboard information in real-time," *2017 4th International Conference on Systems and Informatics (ICSAI)*, Hangzhou, 2017, pp. 1141-1145.
- [8]. Y. Kwon, J Choi, J. Jeon, K. Kim and B. Jang, "Design of Automotive Digital Instrument Cluster Adjustable to Driver's Cognitive Characteristics," *2019 International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju Island, Korea (South), 2019, pp.461-463.