

RESEARCH AND IMPLEMENTATION OF AUTOMATED HIL SIMULATION TEST PLATFORM FOR ECU OF AUTOMOTIVE ENGINE

Lifeng Zheng

Colloge of Vehicle Engineering, Changzhou Vocational Institute of Mechatronic Technology, Jiangsu, China
Email: zhenglifengzlf1982@126.com

Abstract - In order to help automobile factory finding a simple and effective automatic testing equipment to replace the heavy and repetitive manual testing work of test engineers, so that test engineers can concentrate on improving the function and design of test system, firstly, the hardware in the Loop (HIL) simulation test and automation test of the engine electronic control unit (ECU) are deeply studied. Based on the experience accumulated in the past, the possibility analysis and technical evaluation of HIL simulation test, automation test and the real-time requirement of the ECU simulation test are conducted in detail. Then, aiming at the current testing tasks, combined with the commonly used HIL testing methods and technologies in the industry, a new software and hardware platform is put forward to realize the automatic testing program of automotive engine ECU function, which fully meets all the functional and technical requirements of the demand. The results show that the application of this automatic ECU real-time simulation test platform in the development of ECU of hybrid engine proves that it fully meets the actual needs. It can be seen that the ECU development cycle of a hybrid electric vehicle engine will be greatly shortened and the development cost will be saved in combination with the application of the platform.

Keywords: Simulation, Test, Engine, ECU.

1. Introduction

In recent decades, many changes have taken place in the development of automobiles. They have gradually changed from simple mechanical products to intelligent terminals. At the same time, people's demand for entertainment and comfort of automobiles has been increasing. Governments in various countries have put forward increasingly strict requirements on automobile safety and emission. More and more electronic control units (ECUs) have been applied to various control and functional application of automobiles, making the automotive electronic control more and more complex. Its function is more and more complex, and the control software code of a single power system is more than 20,000 lines. Overall, the cost of electronic system and software has accounted for more than 30% of the total vehicle cost, and the higher the high-end vehicle is, the higher the proportion of its electronic system cost is.

Future automobiles may evolve into smart controllers plus wheels, which will inevitably be dominated by high-tech industries, and these core technologies are firmly in the hands of foreign multinational companies, such as BOSCH, DELPHI, DENSO, Continental and other professional automotive technology companies, even IT

technology companies, such as Google, Apple, OnStar and so on. This has also become the biggest obstacle to China's automotive technology progress. If it is impossible to make a breakthrough in the field of automotive information technology, the transformation and upgrading of the automotive industry will never be completed, nor can we compete with foreign automotive manufacturers. However, there are also opportunities. In recent years, the domestic IT industry has become the fastest innovative and most remarkable industry. If the IT talents and IT engineers' cost advantages can be utilized, and the traditional thinking can be completely abandoned, it is bound to make breakthroughs in the core areas of automobiles and usher in the era of our own automobile power [1].

The engine ECU is the most complex and core of the automobile electronic control system and it is the nerve center of the whole automobile. Therefore, the research and development of engine ECU and EMS (Energy Management System) is a key part of the research and development of engine and even power assembly. The hardware development of engine ECU and the realization of engine control strategy and software have occupied a very important position in the development process of new engine. If the engine ECU malfunction will lead to serious consequences, so the engine ECU test work is very important. The control function of engine ECU is also increasing to

meet its requirements, such as small displacement turbocharged engine, hybrid power, compressed natural gas engine, engine start-stop technology and so on [2]. In addition, countries have strict legal requirements on vehicle safety and emission performance, such as Euro IV and Euro V emission standards in Europe and national IV and State V emission standards in China. At the same time, the engine ECU must also realize on-board diagnostic (OBD), error information generation, emergency processing and self-adaptive functions. The diagnostic software code exceeds 50% of the total number of EMS codes, which makes the engine ECU and EMS functions very complex. The complexity of automotive electronic system also leads to the increase of automotive electronic faults. According to the data of German Automobile Association (ADAC), the faults of automotive electronic and electrical system and ignition system, injection system and chassis electronics account for more than 40% of automotive faults, which makes the test of automotive electronic system more important and more complex. Its testing work has also become very complex, and the workload is becoming huger and huger.

In order to reduce the development cost of engine ECU and the risk of product recall, it is urgent to find a simple, fast and efficient simulation test method for engine ECU. In order to save the time and cost of engine ECU simulation test and avoid the damage to testers and engines caused by dangerous working conditions, it is necessary to conduct comprehensive functional simulation and performance test for engine EMS and ECU in the stage of engine hardware development. Therefore, the rapid simulation of engine and the parallel test of ECU become particularly important [3]. In order to meet the requirements, more and more convenient and fast new test methods have been applied to the engine ECU test. Among them, the hardware in the Loop (HIL) real-time simulation test platform test method of engine ECU is the most important.

2. Engine ECU Simulation Test and Flow Chart

The ECU is the core of the automotive EMS, which is assisted by a variety of peripheral sensors and actuators to achieve precise control of the engine [4]. The automotive EMS mainly consists of the following three parts:

Sensor: mainly includes crankshaft speed sensor, camshaft speed sensor, air flow sensor, intake temperature sensor, throttle position sensor, coolant temperature sensor, knock sensor, speed sensor, and oxygen sensor.

Actuator: mainly includes single pump, ignition control module, air flow control module, and various solenoid valves.

ECU: Its function is to collect and adjust the input signals of various sensors, monitor the real-time operating conditions of the engine, including speed, intake pressure, intake temperature, exhaust temperature and other parameters. After processing according to the engine management control algorithm, the control signal is output and power amplification is carried out. Finally, it is transmitted to the actuator to accurately control injection or fuel supply time, fuel injection volume and ignition time, etc. so that the engine has the best power, economy and emission characteristics under various operating conditions.

The engine model is the basis of the whole HIL simulation system. The platform system simulates the static or dynamic characteristics of the engine under various working conditions in real time through Smulink model operation [5]. According to the working principle of each system of turbocharged diesel engine and the clear physical meaning, the controlled object model divides each part into modules to express the comprehensive results of working process and the time average effect of state variables. Differential equation or algebraic equation is used to express the clear physical process, and simple steady-state Map or empirical formula is used to express complex physical process, such as in-cylinder process, which improves the real-time performance of the model. The average value model is modelled by modularization. The model consists of five subsystems: turbocharger model, intake system model, cylinder model, crankshaft dynamic model, and exhaust system model. The intake system model includes intake tube model and intercooler model. Turbocharger model includes turbine model, turbocharger rotor dynamics, and compressor model.

Table. 1 The basic structure and performance parameters of the engine

Name	Parameter	Unit
Cylinder diameter	132	Mm
Piston stroke	145	Mm
Compression ratio	16.5	-
Stroke number	4	-
Rated power	330	kW
Maximum torque	2040	N·m
Effective fuel consumption rate	265	g/(kW·h)

Fuel injection ignition is controlled by Complex Programmable Logic Device (CPLD). CPLD is responsible for the generation of complex current waveform and the fault diagnosis of the actuator. The fault information is sent to the main controller through SPI (Serial Peripheral Interface) interface [6].

The main controller generates fuel injection and ignition pulse signals according to the application requirements and sends them to the IO port of CPLD. CPLD generates synchronous equal length injection current waveform and ignition control signal according to the width of the signal.

When CPLD finds a hardware fault, it only shuts down the output in the current pulse period, sends the diagnosis requirement to the main controller (CPLD interrupt signal PIN32 is set high) and indicates the fault category in the SPI return information. Then, it waits for the processing result of MCU (Micro Controller Unit), and automatically opens the output and updates the diagnosis information at the beginning of the next pulse period [7].

Four sixteen-bit input commands and two sixteen-bit diagnostic data are defined in the process of fuel injection ignition control. Each command has a start mark and corresponding return value, but the return value can only be obtained in the next frame communication. Detailed diagnostic information can be obtained by looking up the table according to the returned data. Figure 1 shows the CPLD control sequence diagram of fuel injection ignition.

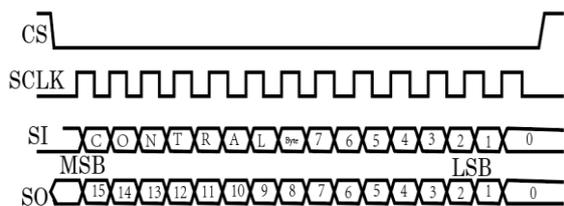


Figure. 1 CPLD control sequence diagram of fuel injection ignition

Throttle drive unit is an integrated intelligent H bridge driver, which integrates diagnosis and protection functions, communicates with main controller through 8-bit SPI interface, and reports diagnosis information. Between the throttle drive unit and the controller, a common IO port DIR (steering) and a PWM port (speed regulation) are used to complete the motor's forward and backward rotation and speed regulation, as shown in Table 2 and Figure 2.

Table. 2 Function value table

DIR	PWM	MODE (Bit2)	OUT1	OUT2	Comments
0	1	0 (slow decay)	H	L	Motor turns clockwise
0	0		H	H	Freewheel with slow decay
1	1		L	H	Motor turns counter-clockwise
	0		H	H	Freewheel with slow decay
0	1	1 (fast decay)	L	L	Motor turns clockwise
0	0		H	H	Fast decay
1	1		H	H	Motor turns counter-clockwise
1	0		L	L	Fast decay

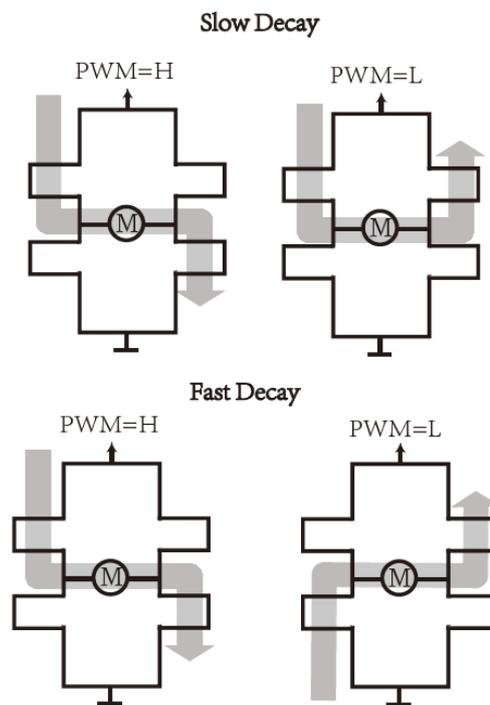


Figure. 2 Principle diagram of DIR/PWM controlled motor's forward and backward rotation and speed regulation

3. Overview of Design of Automated HIL Simulation Test Platform

3.1 Real-time computer

NI industrial computer is used as the underlying hardware platform of the automated HIL test platform. It is interconnected with the upper PC through xPC Network BootImage and converted into a real-time processor. It is also responsible for all task scheduling, including real-time simulation model of running engine, controlled object model of target machine, and control of general board, etc. [8].

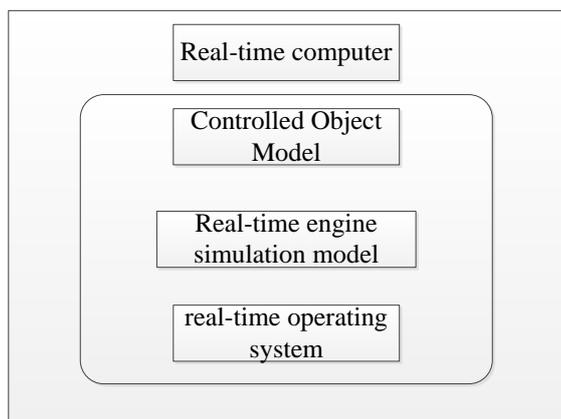


Figure. 3 Real-time computer structural diagram

3.2 HIL Test principle

The real-time processor simulates the various working conditions of the engine to test the ECU comprehensively and systematically. The processor transmits sensor signals needed by ECU, such as crankshaft sensor signal, camshaft sensor signal, temperature sensor signal, detonation sensor signal, oxygen sensor signal, air flow sensor signal, etc. These signals are sent to ECU signal entry through IO card conversion and signal conditioning. After receiving the sensor signal, ECU calculates the execution signal according to the control strategy, such as throttle control signal, fuel injection ignition control signal, valve control signal and so on. HIL simulation test platform receives these actuator signals through IO card. After conversion and signal conditioning, they are transmitted to the engine simulation model to achieve the closed-loop test. At the same time, HIL simulation test platform controls and monitors the whole closed-loop through its abundant functions of data acquisition and transmission, CAN (Controller Area Network) message monitoring and fault simulation. The platform can also realize the automation of all simulation and test processes through the host computer automation test management software.

The HIL simulation test platform of engine ECU is based on a real-time processing computer, which connects the host computer with the closed-loop

equipment of the ECU under test. Real-time computer runs engine real-time simulation model, host computer runs target computer controlled object model, and HIL simulation test platform is connected with measured ECU through a variety of real-time IO processing boards. It simulates actual engine operation through engine model, and realizes real-time interaction between control and sensor signals with engine ECU.

The real-time processor simulates the various working conditions of the engine to test the ECU comprehensively and systematically. The processor transmits sensor signals needed by ECU, such as crankshaft sensor signal, camshaft sensor signal, temperature sensor signal, detonation sensor signal, oxygen sensor signal, air flow sensor signal, etc. These signals are sent to ECU signal entry through IO card conversion and signal conditioning. After receiving the sensor signal, ECU calculates the execution signal according to the control strategy, such as throttle control signal, fuel injection ignition control signal, valve control signal and so on. HIL simulation test platform receives these actuator signals through IO card. After conversion and signal adjustment, they are transmitted to the engine simulation model to achieve the closed-loop test. At the same time, HIL simulation test platform controls and monitors the whole closed-loop through its abundant functions of data acquisition and transmission, CAN message monitoring, and fault simulation. This platform can also realize the automation of all simulation and test processes through the host computer automation test management software [9].

HIL simulation test platform includes real-time computer, engine simulation model, target computer controlled object model, and thousands of signal conversion and conditioning boards, and its functions are very complex. In order to understand the functions of HIL simulation test platform more clearly and clarify the relationship between various functional modules, and also to reduce the difficulty of developing automated HIL simulation test platform, the platform is independent to host computer, power management, programmable power supply, controller to be tested, power supply module of controller, fault injection module, load simulation module, real components, interface adaptation, signal conditioning, and real-time computer.

The whole automated HIL simulation test platform is divided into two parts: host computer and target machine. The host computer runs the test management software, automated test management software, and Matlab/Simulink assistant software, which are windows for the interaction between the automated HIL simulation test platform and the tester. The tester controls the target computer through the controlled object model of the target computer and runs on it.

The test management software and automation test management software in the host computer realize the test and automation test of engine ECU project. Target machine is a complex system based on real-time computer, which runs the controlled object model of target machine and integrates power control unit and signal conditioning and conversion unit. Among them, the power control part configures the control voltage and current according to the test management software of the host computer to supply power for the real-time computer and the controller to be tested, and at the same time realizes the power related test. Signal conditioning and conversion components include several functional modules such as signal conditioning, interface adaptation, fault injection, load simulation, and real load. It collects, adjusts, converts and transmits all electrical and digital signals.

3.3 Overall architecture of automated HIL simulation and test platform

The automated HIL simulation test platform consists of host computer and target machine. The host computer is responsible for the manual and automatic control of the whole management process and the monitoring of the interaction process between compiling, downloading and engine ECU. The target machine is composed of real-time computer and rich IO boards and other hardware devices [10]. It mainly runs the real-time simulation model of engine and the controlled object model of the target machine, and regulates and monitors the signal of interaction between engine ECU.

In the specific scheme, xPCTarget dual-computer system is used to realize the connection environment between host computer and target machine. MATLAB/Simulink is regarded as the realization tool of target machine controlled object model, and Python language is used to write

automatic test program and realize automated simulation test.

Among them, the host computer and its running test management software, automated test software, and Matlab/Simulink software are the core of the whole HIL simulation test. The test case is executed by test management software, and the engine simulation model is controlled to simulate various working conditions of the engine. At the same time, the interaction process between the engine simulation model and the engine ECU under test is monitored, and the functions of the engine ECU under test are tested. The automated testing software is used to realize the automated testing of all test items and generate documents such as automated test cases and reports.

Target machine is based on real-time computer. It integrates power management module, interface adaptation, signal conditioning, fault injection module, load simulation, and real load and other hardware devices. The real-time simulation model of engine running in real-time computer is used to simulate the real engine running, and the control strategy of engine ECU under various working conditions is tested. Figure 4 shows the overall architecture of the automated HIL simulation test platform.

By controlling the controlled object model of the target machine running on the real-time computer by the test software in the host computer, the test functions of the engine ECU can be realized according to the pre-set test cases, and the functions of the engine ECU can be tested. At the same time, the host computer accesses the EMS software variables running in the engine ECU through the calibration protocol, and verifies the control strategy of starting the ECU by monitoring and modifying the variables.

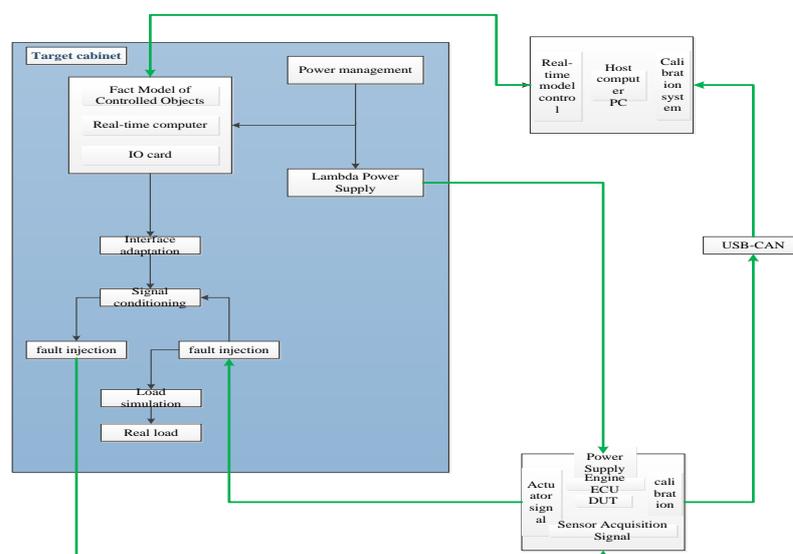


Figure. 4 The overall architecture of automated HIL simulation and test platform

The host computer is connected with the target machine in the form of xPCTarget. xPCTarget is a Real-TimeWorkShop tool provided by MathWorks. This tool can transform standard PC hardware into real-time processor, which can be used as a real-time test environment. The connection between xPCTarget Network Boot Server and xPCTarget Network Boot Target PC is established through Network Boot Image, in which xPC Target Network Boot Server is used as a real-time test environment. xPC Target Network Boot Server is the host computer, and xPC Target Network Boot Target PC is the real-time computer in the target machine. In this dual-computer mode, the controlled object model of the target computer can be compiled and downloaded to run in the target computer [11]. At the same time, because of the perfect support of MATLAB/SIMULINK software in the dual-computer environment of xPC Target, MATLAB/Simulink can be used to develop the controlled object model of the target machine. Moreover, xPC Target dual-computer environment supports board driver written in C language. The host computer and the target machine can be connected through Ethernet. After the underlying environment is built, it is possible to access the target machine remotely through Ethernet. In this way, remote automated simulation testing can be achieved.

After the upper computer is connected with the real-time computer environment, the host computer downloads the controlled object model of the target machine to the real-time computer, and controls the parameters of the engine simulation through the real-time control model to simulate the operation of the engine. Controlled object model is based on MATLAB/Simulink. MATLAB can provide functions such as calculation and visual programming. It provides the ability to solve real problems by mathematical methods without using traditional programming. Simulink is an interactive environment for simulation, modeling and analysis of various dynamic models. It provides an easy-to-use visual graphical interface, which is very convenient to establish the system. The controlled object model can be easily created through the rich functional blocks provided by Simulink, without having to write all the code by hand. The controlled object model of the target machine is essentially a bridge connecting the IO card and the real-time simulation model of the engine in the target machine to the control model of the host computer. Through the controlled object model of the target machine, the host computer can control the input and output of the IO card, the parameters of the real-time simulation model of the engine, the input and output variables of each card, and the parameters of the engine simulation model, etc. After building and compiling the controlled object model of the target machine, it is downloaded to the target machine, so that the hardware and software connection between

the host computer and the target machine is established. At the same time, in the compilation process of the controlled object model of the target machine, a mapping file, XML (Extensible Markup Language) type between the resources and addresses of the controlled object of the target machine can be generated, which describes the logical address of the resources and variables in the target machine accessed by the host computer.

At the same time, the EMS variables in the ECU of the tested engine can be collected in real time by the calibration system running on the host computer, and the EMS variable values of the tested engine can be modified in real time according to the requirements of the test cases. The calibration system is based on the common CCP (CAN Calibration Protocol) protocol. Firstly, it is necessary to write the calibration code according to CCP protocol in the development of ECU software. In the code compilation stage, a2l files mapping variable names and variable addresses can be generated. The calibration system accesses the variables in the ECU of the engine under test through a2l files. Through the calibration system, the host computer can monitor the internal variables of the engine ECU in real time and modify them in real time. This not only can monitor the control logic of the engine ECU under test, but also verify its control strategy through real-time modification of the internal variables of the engine ECU under test. Vector's CANape is used as a monitoring tool to calibrate the CAN bus in the system.

Because the development task of the automated HIL simulation test platform is very large, the whole software development task is divided into three independent tasks: driving implementation and model establishment, establishing test engineering and connecting PC with target computer, calibration mapping, and calibration system establishment. Figure 5 is the development task flow chart of the HIL simulation test platform. These three independent tasks can be accomplished in parallel, which can not only shorten the development cycle of the whole platform greatly, but also make the development tasks and objectives clearer, and facilitate project managers to better grasp the development progress.

Firstly, it is necessary to select a series of hardware devices, such as IO card and signal conditioning board, according to the needs of the project. The process of hardware selection is a very complex process, which directly affects the performance of the whole HIL simulation test platform. Then, because there is no underlying driver used in MATLAB/Simulink, it is supposed to implement the driver of the board itself, including writing, compiling and encapsulating the card driver code. The card driver is implemented in C language, and the driver is compiled and encapsulated through Simulink. When all board drivers are implemented,

the controlled object model of the target computer is built in MATLAB/Simulink, including the allocation of board resources, the correspondence between the controlled object model and hardware equipment, and the establishment of the model. This model is the interface of the host computer to access the resources of the target machine. The XML file generated by the compiler model is a mapping file between resources and software variables, and is the interface between hardware and software.

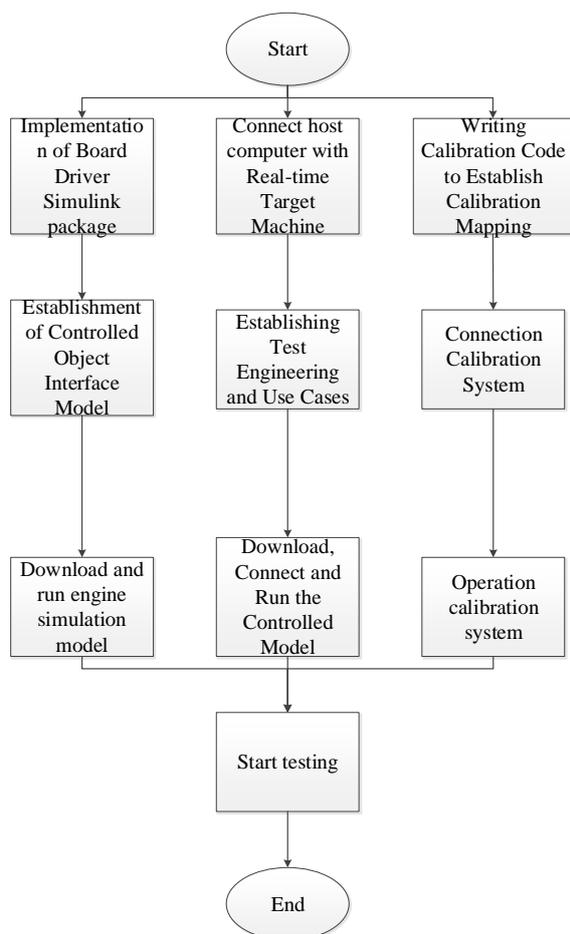


Figure. 5 Development task flow chart of automated HIL simulation test platform

Secondly, it is necessary to establish the connection between xPC Target Network Boot Server and xPC Target Network Boot Target PC through Network Boot Image, so that the connection between host computer and target machine can be established. Then, a complete set of test cases are designed to ensure the high coverage and scientific nature of the test. At the same time, a test project is set up in the host computer according to the test requirement. The test project is composed of graphical interface, which integrates virtual instrument, controlled object model, and calibration system. In this project, the tester can download, connect and run the controlled object model only by clicking the controlled object model button. Through the corresponding virtual instrument, the

corresponding control module can be controlled to make the corresponding response, and IO, AD, PWM, fuel injection ignition and other functions can be intuitively tested. Meanwhile, in the test project, the variable values in the ECU of the engine under test can be monitored and modified in real time by dragging and dropping the variable names in the calibration system into the Input/Output monitor window. It can operate various functions or hardware devices such as FIU, Watch Windows, and realize the test of related functions. The purpose of this test project is to facilitate test engineers to test better. It is a human-computer interaction interface for test management of HIL simulation test platform.

Finally, the implementation of the calibration system can also be carried out in parallel. The relevant calibration functions are defined according to CCP protocol during the development of engine ECU. In the process of compiling ECU code, a2l file is generated. This file is the mapping file of the internal variables and addresses of the engine ECU under test. The variable names in the engine ECU under test correspond to the variable names in the calibration system. The host computer connects the communication CAN bus of the ECU of the tested engine through CANape, so that the calibration system can monitor the CAN bus message in real time through CANape, and the calibration system can also modify the internal variables of the ECU in real time through CAN message and variable name.

Firstly, the principle of HIL testing is explained, and the whole system is divided into several relatively independent modules to improve the independence, forming a high cohesion and low coupling host computer, power management, programmable power supply, controller to be tested, power supply module, fault injection module, load simulation module, real components, interface adaptation, signal conditioning, real-time computer 11 function modules. It not only reduces the difficulty of implementation, but also facilitates the daily operation and maintenance of the platform. Through the analysis and design of each independent function module, the data and control relationship between each module is sorted out, and the data type and structure of each module and the interface of control information are abstracted.

Finally, the overall architecture of the automated HIL simulation and test platform is briefly introduced. The hardware, software and technical methods used in the platform are described, and the workflow of each part is explained. At the same time, the whole platform implementation process is divided into three parallel implementation processes, and the three processes are explained. This not only can shorten the development cycle of the whole platform, but also make the content of each development task and development objectives clearer, and facilitate project managers to better grasp the overall development progress.

3.4 Base architecture and software of automated HIL simulation and test platform

As the core of the car, the quality of the engine affects the quality of the car and the evaluation of the car by consumers, and even the reputation of the car brand. Therefore, as the last guarantee of quality management before engine leaving factory, HIL test of engine is particularly important. For HIL test of ECU, the hardware quality, reliability, stability, coverage and scientificity of test scheme of HIL simulation test platform are the key factors affecting the test results. Firstly, the hardware of the test platform must meet the real-time and reasonable performance requirements of the test requirements. Generally speaking, the processing signals of engine ECU are divided into sensor signals and actuator signals. In the process of HIL simulation test of engine ECU, the most important thing is the real-time simulation of engine and the real-time processing of engine ECU signals. That is to say, the hardware of HIL simulation test platform and the real-time requirement of engine simulation model are very high. This is also the key technology and technical problem in HIL testing. This requires using high-performance real-time processor, efficient and reliable IO card, and real-time operating system to run engine simulation model and controlled object model of operating target machine.

After several discussions, the underlying hardware is divided into six parts: system power supply, board, interface adapter circuit board, load board, interface circuit, and industrial computer. Among them, system power integrated power management, programmable power supply, and controller power supply module are three functional modules; load board integrated load simulation module and real component two functional modules; board card integrated signal conditioning functional modules; integrated fault injection module for adapted circuit board.

The underlying hardware mainly includes two parts: system power control and signal processing. The system power supply control includes three functional modules: power management, programmable power supply, and controller power supply module. It needs to supply power to real-time computer, adapter circuit board and ECU under test. The power management module realizes the power management of the HIL simulation test equipment and it is the entrance of AC (Alternating Current). It mainly realizes the functions of AC on-off control, overload protection, and emergency stop control.

Because it needs to simulate the power supply when the car is running, the commonly used Lambda programmable power supply is chosen, which can control the output voltage of the power supply through the pre-set program or waveform. The power supply module of the controller is mainly responsible for monitoring and controlling the voltage and current of the ECU under test and feeding back to the host computer as the return value of the test.

The signal processing part is mainly composed of real-time computer, board, adapter circuit and interface circuit. The controlled object model and real-time engine simulation model are run on the real-time computer. The engine simulation model simulates the real engine operation and interacts with the ECU under test. The sensor signals transmitted by the engine simulation model are transferred to the engine ECU through board adjustment and adapter circuit conversion. Engine ECU transmits actuator signal through internal control strategy, and then transfers it to engine simulation model through board adjustment and adapter circuit conversion. The adapter circuit includes a series of acquisition and conversion components, such as transformer conversion, DMM (Digital Multi-meter) sampling multiplexer switch, digital switch output, acquisition window, etc., to convert signals. Card components include AD, DA, AI, AO, DI, DO and other cards, and their corresponding digital and analog signal acquisition, transmission and conversion functions. At the same time, it includes CAN and LIN interfaces, which can receive and send CAN and LIN messages directly.

The system power supply includes three functional modules: power management module, programmable power module and controller power supply module. It supplies power for real-time computer and adapter circuit, and supplies power for the ECU under test through interface circuit. The power management of HIL simulation test equipment is realized by power supply components, including on-off control of power supply, overload protection, emergency stop control, alternating current configuration, power consumption control of equipment itself and other functions.

Considering the comprehensive performance and cost, the knob switch, air switch, overload protection switch and emergency stop switch which are common in the market are chosen.

At the same time, Lambda power supply is selected to meet the test requirements of the ECU under test, such as the test and automatic configuration of power supply shock, high and low

voltage, and voltage fluctuation energy conditions. The voltage and current of the power supply used to simulate the actual operation of the controller can be controlled manually or through the output port of the real-time computer, especially the simulation of the vehicle power supply shortage, the impact of engine start-up on the power supply and other conditions. Lambda power supply has memory function.

Even after power off, it automatically adjusts the voltage and current to the last shutdown, and can connect the host computer through Ethernet, so as to realize remote and automatic control of power supply.

3.5 Automatically implementing HIL test

The automatic test program is written in Python language, the test cases are executed automatically, and the controlled object model and the calibration system of the target computer are controlled in real time. Thus, the closed-loop simulation test system can be realized, namely, host computer of the automatic HIL simulation test platform - the controlled object model of the target computer - the target computer of the automatic HIL simulation test platform - the ECU of the engine under test - the calibration system - the host computer of the automatic HIL simulation test platform.

The automated HIL simulation test program does the same work as the manual HIL simulation test introduced before, but it works in different ways. The manual HIL simulation test introduced in the previous section is completed manually by the test engineer through the test software of the host computer, while all the work in the automatic HIL simulation test is completed by the automated test program. That is to say, test cases and test schemes established by test engineers must be implemented in Python language. It includes testing methods, test logic, test execution sequence, test result analysis, access to controlled object model of target machine, access to calibration system, generation of test case description and test report, etc.

The automated HIL simulation test program realizes the automatic test of fuel injection ignition, detonation, power supply, AD, DA, AI, AO, DI, DO, DMM, eTPU, PWM and other modules. Next, taking the automatic test process of analog output signals as an example, the following processes are described in detail: writing test cases using Python, automatic generation of test case documents, automatic test flow control, automatic access to the controlled object model of the target machine, automatic access to calibration system, automatic test result analysis, and automatic generation of test reports.

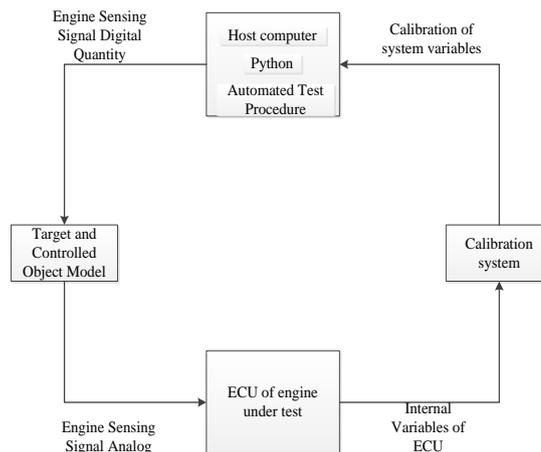


Figure. 6 AO automated test block diagram

In AO test, the automatic HIL simulation test platform can access the controlled object model of the target machine and send the sensor analog signal to the ECU of the engine under test by executing the automatic test program. The ECU of the engine under test receives the sensor analog signal and acts accordingly. The host computer reads the corresponding variable value of the ECU of the engine under test through the calibration system compared with the transmitted signal value, to judge whether it conforms to the control logic and automatically analyses the test results. The test results are given in the form of word version test report [12]. In summary, the key to realize automated AO testing is to write automated testing program through Python, so as to realize real-time access and control to the controlled object model and calibration system of target machine.

3.6 Automated test

For the automated testing of AO signals, the test case is actually executed by the host computer automated testing program, and a sensor signal is sent to the engine ECU under test through the controlled object model of target machine. The engine ECU under test receives the sensor signal and then acts accordingly. The host computer reads the variable values related to the sensor signal in the engine ECU through the calibration system, which are compared with the set sensor signal values. In accordance with the test logic, it is possible to determine whether the channel works properly.

After the host computer achieves real-time access to the controlled object model and calibration system of the target machine through the automated test program, it only needs to control the internal variables of the program according to the test case, such as setting the variable `app1` to 3.00, and comparing the value of the read variable `CPP.AD_App1` with 3.00. If the error range is not

more than 5%, the test item will succeed; otherwise, the test will fail.

Figure 7 is the flow chart of AO class signal automation test program. The program is divided into six parts: establishing test cases, executing test cases, accessing the controlled object model of the target machine to send the corresponding sensor signals, accessing the calibration system to read the

corresponding signals in the ECU of the engine under test, comparing the sensor signals and the values of variables in the ECU to determine whether the engine control logic is satisfied, automatically generating test reports, and automatically generating test case description documents. The six parts will be described in detail.

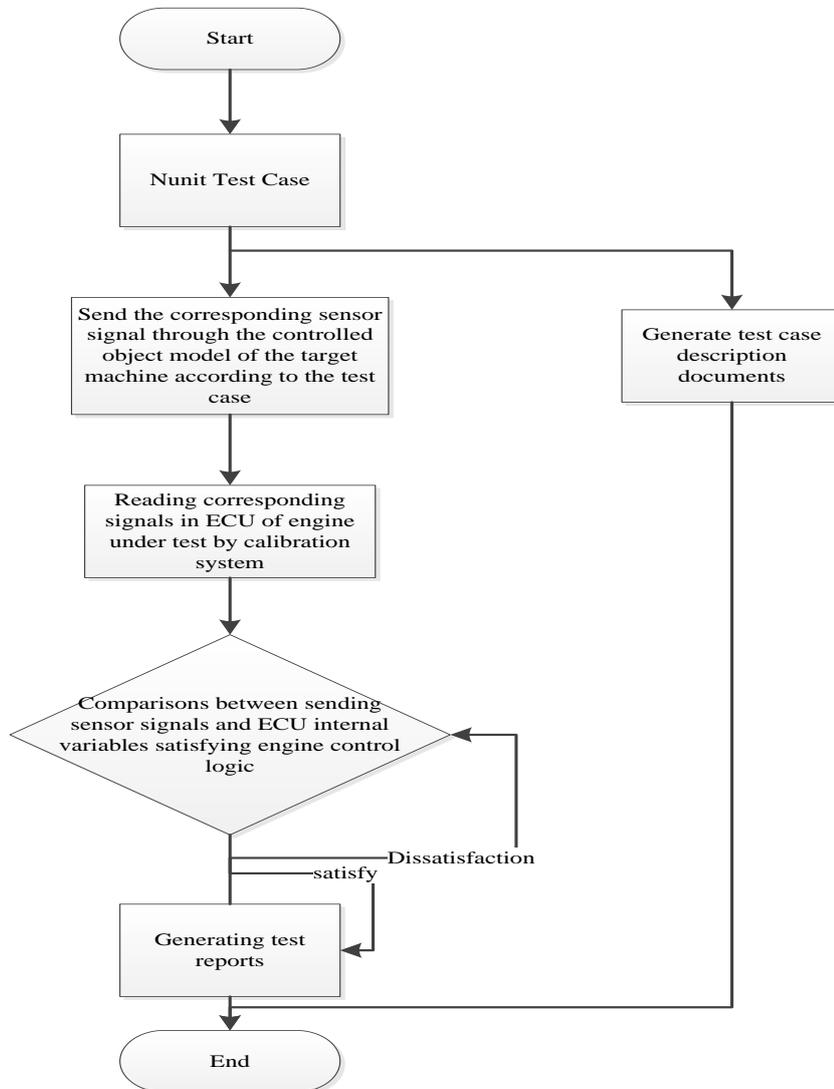


Figure. 7 The flow chart of AO class signal automation test program

3.7 Introduction to Python Language

Python was invented by Dutch Guidovan Rossum at the end of 1989 and its first official version was officially released in 1991. It is a clear object-oriented language with simple features, open source, cross platform, object-orientation, and rich standard libraries.

Python grammar is very simple. It uses indentation as the specification of statements, which makes the grammar of Python program very strict, but also guarantees the readability and maintainability of the program.

The underlying data structure includes string, list, and dictionary, which are easy to use. Openness is Python's greatest advantage. Users can get all the source code of Python language free of charge and modify it at the same time. It has been greatly developed because of its open source and free characteristics. Python language can adapt to cross-platform operation very well. For this reason, Python is also called "glue". Through Python, programs or software such as C, C++, Java, XML, dynamic library files, COM objects can interact smoothly. Python also provides interfaces to most databases to facilitate users' access to databases.

It is an object-oriented language, which perfectly supports all inheritance, encapsulation, polymorphism and other object-oriented attributes. Python has a very rich standard library and third-party software modules and facilitates users to achieve its functions quickly and efficiently through the standard library. Python is becoming more and more important in the field of programming languages. Companies like Microsoft, CG and Google use Python extensively.

The software and hardware of the automated HIL simulation test platform come from different vendors, and their interfaces are different. Therefore, a simple and cross-platform language is needed to realize its functions. It is also because of Python's advantages of cross-platform and abundant third-party module library support, Python is chosen as the development language for the host computer automation test software of the automated HIL simulation test platform.

The controlled object model of the target machine is the medium of access and control between the host computer and the target machine. Similarly, it is necessary to access the controlled object model of the target machine in automated test. Different from the way in which the tester accesses the controlled object model of the target machine through virtual instrument in manual test, the work in automated test must be completed automatically by the automated testing program. It requires that the automated testing program accesses and controls the controlled object model of the target machine through Python.

4. Conclusion

Automotive engine ECU as the core of the car, its importance is self-evident. And its testing is also a key link in the development of automotive engines. However, its testing process is complex and cumbersome. Traditional testing methods and techniques cannot meet the requirements of modern testing. Since its emergence, HIL simulation test of engine ECU has rapidly developed into the most popular test and assistant development method with its advantages of high efficiency, speed, safety and low cost. After consulting a large number of relevant materials and documents, and doing a lot of research on engine testing technology and methods, relying on the rich experience of engine ECU testing in FAW Technology Center of China and the solid theoretical foundation of Jilin University Intelligent Control and Embedded System Laboratory, a complete implementation scheme of an automated HIL real-time simulation test platform for engine ECU is proposed, and the development of the platform is successfully completed. At the same time, the

platform is applied in the development and testing process of several engine ECUs. After the verification and testing of the actual test work of several engine ECUs, the platform can accomplish the testing work well, greatly shorten the development and testing cycle of the project, avoid the damage that may be caused by dangerous working conditions, and save a lot of manpower cost for the project development.

The main work is as follows:

After consulting a large number of test signal processing related data and IO card instruction manual, the selection of general IO card and signal conditioning card and the compilation, compilation and packaging of some board bottom drivers are completed.

According to engine ECU test requirements, test board resources are allocated and hardware and software resource mapping tables are established. Based on the resource mapping table, the controlled object model of the target machine is established, compiled, downloaded and run by using MATLAB/Simulink.

By consulting the relevant literature and information, the dual-computer model of xPC Target is proposed, and the dual-computer environment of host computer and target machine is realized. The host computer accesses and controls the target machine through the real-time controlled object model of target machine.

Establish test engineering, including design of test strategy, test cases, etc., build virtual instrument according to test requirements, load real-time controlled object model of running target machine, and connect calibration system. It is applied to the actual development and test project of an engine ECU. The engine ECU can test the functions of fuel injection ignition, detonation, power supply, AI, AO, DI, DO, eTPU, PWM and so on.

Python language is used to write automated test program to realize the automated test of all test items, as well as the automatic generation of test case description documents, test reports, test results specifications and other documents. It is also applied to the actual engine ECU development and test project.

The automated HIL simulation test platform is used in the test phase of several engine research and development projects in FAW Technology Center to realize the test work. The test functions, reliability and stability of the platform are verified.

At the same time, HIL simulation testing technology has obvious advantages in repeatability, instantaneity, working condition comprehensiveness and the convenience of ECU development, and has been paid more and more attention to. Aiming at improving scalability, reducing cost, enhancing reliability and maintaining the continuity of platform development, the overall design scheme and basic

requirements of HIL simulation platform based on rotational speed simulator and ECU hardware-in-the-loop simulation platform are proposed by analyzing the working principle of HIL simulation system and the actual needs of the laboratory. The operation principle, hardware development, engine average model, and a series of development links, such as the design of relevant signal processing model, are studied.

Acknowledgement

Research startup subject of Yangtze Normal University: 2017KYQD16.

References

- [1] Li, H., Huang, Y., & Zhang, F. (2016). Design and implementation of diesel engine ECU software based on software components library. *Automotive Engineering*, 38(12), 1420-1426 and 1419.
- [2] Ham, W. K., Ko, M., & Park, S. C. (2017). A framework for simulation-based engine-control unit inspection in manufacturing phase. *Control Engineering Practice*, 59, 137-148.
- [3] Hideki, Y. (2017). Valve control device and valve control method for internal combustion engine. *Toyota Jidosha Kabushiki Kaisha*, 29(6), 658-64.
- [4] José, A., Valencia, C., Adalberto, G., & Díaz, T. (2017). Simulation NMPC in 2-HIL to design ECU. *International Journal on Interactive Design & Manufacturing*, 11(4), 859-879.
- [5] Grasreiner, S., Neumann, J., & Wensing, M. (2017). Model-based virtual engine calibration with the help of phenomenological methods for spark-ignited engines. *Applied Thermal Engineering*, 121, 190-199.
- [6] Buck, R., Damitz, J., & Samuelsen, D. (2017). Method and apparatus for controlling an internal combustion engine. *Journal of the Acoustical Society of America*, 73(6), 1410-1410.
- [7] Scalable. (2016) and efficient configuration of time-division multiplexed resources. *Journal of Systems and Software*, 113, 44-58.
- [8] Yonglin, B. (2015). Design of a ground HIL simulation test platform for flight control system of UAV. *Journal of Terahertz Science & Electronic Information Technology*, 13(6), 903-907.
- [9] Hartavi, A. E., & Uygan, S. M. C. (2016). A hybrid electric vehicle hardware-in-the-loop simulator as a development platform for energy management algorithms. *International Journal of Vehicle Design*, 71(1/2/3/4), 410.
- [10] Atlas, E., Erdoğan, Melike. Irem., & Ertin, O. B. (2015). Hardware-in-the-Loop Test Platform Design for UAV Applications. *Applied Mechanics and Materials*, 789-790, 7.
- [11] Alberto, T. D. L. C., Riviere, P., Marchio D. (2017). Hardware in the loop test bench using Modelica: A platform to test and improve the control of heating systems. *Applied Energy*, 188, 107-120.
- [12] Xiong, R., Duan, Y., & Cao, J. (2018). Battery and ultracapacitor in-the-loop approach to validate a real-time power management method for an all-weather electric vehicle. *Applied Energy*, 217, 153-165.