A Low Cost, Portable Engine Electronic Control Unit Hardware-in-the-Loop Test System

A. Çebi*, L. Güvenç*, M. Demirci*, C. Kaplan Karadeniz**, K. Kanar** and E. Güraslan**

* Automotive Controls and Mechatronics Research Center, Department of Mechanical Engineering, Istanbul Technical University, Istanbul, Turkey

** Product Development Division, Engine and Powertrain Group, Ford Otosan, Kocaeli, Turkey

Abstract— **Engine management and control systems are important components of present day road vehicles as they have a vital impact on fuel economy and reduced emissions. Engine developers want to run the ECU while monitoring its internal state through special programs, after changing part of the ECU code or tables in it, to make sure that everything is in order. Hardware in the loop test systems are used to supply the necessary sensor signals to the ECU during such tests. This paper presents a low cost, portable engine ECU hardware-in-the-loop simulation system that uses a PC card in a laptop computer for supplying the necessary crank and camshaft sensor signals. The signals are generated in a Simulink® program which can also accommodate simple engine models.**

Keywords: hardware-in-the-loop, engine ECU, real time simulation, cam and crank signal generation.

I. INTRODUCTION

Automotive control and mechatronics is increasing its presence and importance every day in today's road vehicles. The road vehicle is becoming a totally mechatronic system loaded with a large array of sensors and actuators monitored and controlled, respectively, by a large number of electronic control units in a vehicle wide network. Engine control was one of the first successful applications of automatic control in road vehicles. Consequently, complicated electronic engine control systems have been around for decades. Their complexity is steadily increasing due to increased levels of conflicting demands on allowed undesired emissions, fuel economy and comfortable operation. The increased complexity of engine control systems is made possible by advances in embedded control hardware and better use of advanced control methods in implementation.

Engine management and control systems exist as code in an engine Electronic Control Unit (ECU) in production vehicles. The ever increasing demand to reduce product development times also applies to the engine management and control system development for a new engine or for an engine being adapted to a new vehicle. Road testing and dynamometer testing are very useful and indispensable tools but the need for faster development times coupled with the desire to test conditions that may not easily be created otherwise have necessitated the development of hardware-in-the-loop (HiL) testing of engine ECUs. Examples of engine ECU HiL test systems can be found in the references [1-4].

This paper presents initial results on the development of a low cost, portable engine ECU hardware-in-the-loop simulation system. The system uses a National Instruments DAQCard 6024 PC card in a laptop computer for supplying the necessary crank and camshaft sensor signals. The signals are generated in a Simulink® program which can also accommodate simple engine models. The Real Time Workshop® and Real Time Windows Target® options of Simulink® are used to generate and execute the hardware-in-the-loop test program in real time. Experimental results are used to demonstrate the effectiveness of this approach. The end result is a low cost and portable engine ECU hardware-in-the-loop test system.

The organization of the rest of the paper is as follows. Motivation and background information are given in section 2. The portable engine ECU HiL test system is presented in section 3. Crankshaft and camshaft position sensor signal generation is presented in section 4. HiL simulation test results are given in section 5. The paper ends with conclusions and recommendations given in the last section.

II. MOTIVATION AND BACKGROUND

A current low cost solution to engine ECU HiL simulation is to use a so-called 'laboratory car' or lab car. A lab car is an electronic device that generates the signals that an engine ECU would normally receive from the engine sensors and other subsystems of the vehicle during operation. A typical electronic lab car is not a very expensive device but it suffers from the necessity of having to manually adjust important variables like engine speed and pressure and temperature sensor values by turning knobs during HiL testing.

The work reported here is part of a larger, comparative study involving a range of hardware and software platforms that can be used for engine ECU HiL tests. The first aim of the study was to increase the applicability of a lab car by adding programmable engine speed capability. This was achieved by generating the necessary crankshaft and camshaft position sensor signals using the HiL system presented here. The second aim was to prepare a program for injecting crank and cam sensor signal errors to the ECU to monitor correct error handling of the ECU in the case of erroneous signals. The target application is to help engine product development engineers who test changes in ECU code and tables by letting the ECU run for several hours or days while they monitor its state using a program

0-7803-8738-4/05/$20.00 ©2005 IEEE

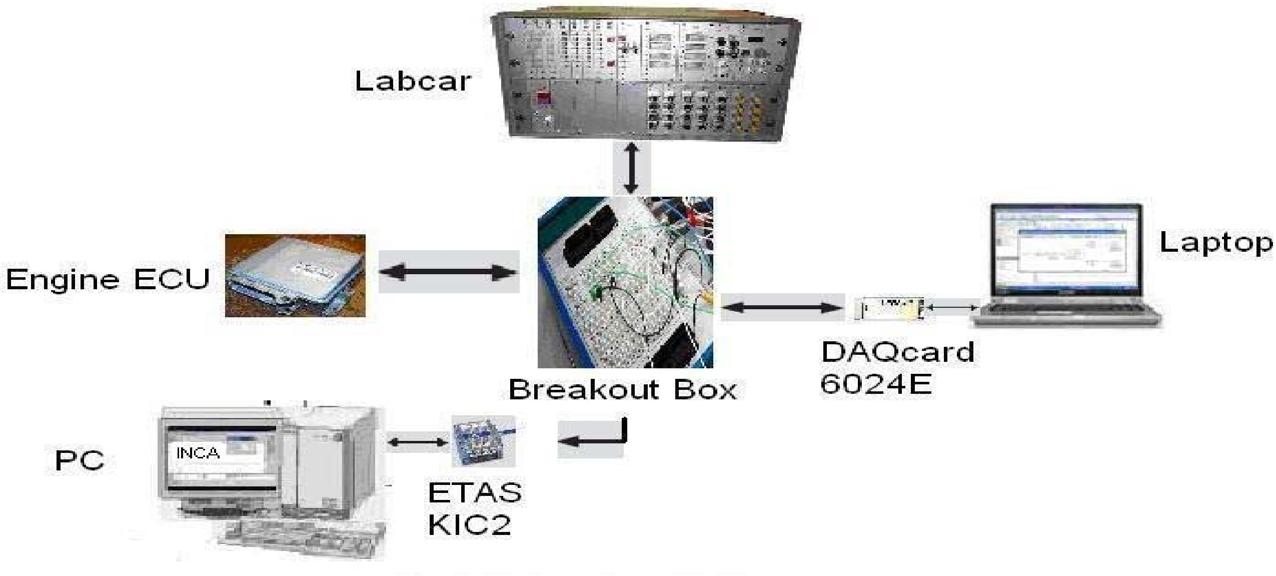

Figure 1. Hardware architecture of the HiL test system

like INCA. The second target application is for checking the effect of crank and camshaft position sensor faults on ECU operation.

The ECU used in the experimental tests was a Bosch ECU for a six cylinder diesel engine with a common rail and turbocharger. The authors actually tested five hardware platforms (cards or systems used for signal generation and data acquisition) and three software platforms (real time operating system) as part of a larger effort [7]. The hardware platforms used were:

- The laptop plus DAQCard 6024E PC card system (present paper),
- A PC plus an NI6025E PCI card,
- An xPC® targetbox and a host PC,
- A dSpace DS1103 card in an expansion box and a PC,
- A dSpace compact simulator comprising of DS1005 processing and DS2210 signal generation cards and its connection box.

The software platforms used as the real time operating system all relied on the Simulink Real Time Workshop® for automatic C code generation and were:

- The Real Time Windows Target®,
- The xPC Target®,
- The dSpace RTI (Real Time Interface).

Along with the crank and camshaft position sensors, five other sensor signals were determined to be important. They are the oil pressure, boost pressure, rail pressure, coolant temperature and boost temperature. These pressure and temperature sensors were modeled as look-up tables. Throttle position sensor inputs could be supplied to the ECU. Also, the six injection pulses sent by the ECU were captured and recorded for later analysis.

The dSpace compact sized simulator and the DS1103 board by dSpace were found to be the most capable solutions. In contrast, the hardware (PCMCIA card in laptop) and software platform (Real Time Workshop® plus Real Time Windows Target®) used in this paper were the cheapest, most portable and admittedly the least capable solutions among all of the systems considered and tested. However, engine product development engineers have found it to be a useful tool when used with an electronic lab car (another cheap solution) or alone for running their ECU with code changes for several hours or days to monitor the internal signals of the ECU using the program INCA and making sure that the ECU system does not hang during operation.

Obviously, a PC equipped with Matlab/Simulink® and the Real time Workshop® can generate camshaft and crankshaft position signals, and the ECU can reconstruct engine speed from these signals. On the other extreme, it is also possible to just buy an expensive HiL simulator which will work quite satisfactorily. The solution proposed in this paper cannot replace an expensive HiL simulator but is, nevertheless, a low cost alternative, worthy of consideration for less demanding HiL tasks. It should also be noted that the very essence of HiL simulation is actually generation of the crank and camshaft position sensor signals especially with the added capability that they correspond to an engine speed that changes in time based on the user input (programmed as a waveform previously or entered on the fly).

III. PORTABLE ENGINE ECU HIL TEST SYSTEM

Automotive companies typically change the calibration tables in the engine ECU and sometimes replace the ECU supplier's default code for a specific control task with their own proprietary software. As such, they are interested in testing this code by running the engine ECU in a HiL setting for several hours or for days to determine and correct potential problems before testing on the actual engine on a dynamometer or on the actual vehicle. Complicated ECU HiL test systems, where each important signal physically connected to the ECU can be varied in a programmable manner, are quite costly solutions with an order of magnitude difference in cost as compared to the simple electronic lab car. Low cost, portable solutions that enhance the capabilities of an electronic lab car by making some of the key signals programmable are therefore quite useful, intermediate solutions.

All engine ECU HiL simulators basically need a computer for programming, visualization and analysis of results and a signal generation card for generating the signals required by the engine ECU for operation. Easy portability of an engine ECU HiL simulator requires the use of a laptop as the computer. Low cost is achieved by using a low cost system that can generate the most critical signals.

The architecture of the portable engine ECU HiL test system that is presented in this paper is shown in Figure 1. In this architecture, an electronic lab car is connected to an engine ECU through a breakout box. A laptop with a PC card size data acquisition and signal generation card (DAQCard 6024E PC card) is also connected to the engine ECU through the breakout box. The ECU K-line is connected to a PC through an ETAS KIC2 connection. The software program INCA in this PC is used to monitor the state of the ECU (engine speed, cam segments, crank and camshaft status, error codes, pressure and temperature values etc.) during operation. Note that the K-line connection and INCA program could also have been installed on the laptop used for the HiL simulations.

By proper manipulation of the switches in the breakout box, the electronic lab car supplied versions of key signals like crankshaft and camshaft sensor signals can be by-passed and these signals can be supplied by the laptop based HiL simulation system. The test results that will be presented in this paper are generated using the DAQCard 6024E PC card with two 12 bit analog outputs. The HiL simulation capability from the laptop based simulator was therefore limited to two signals which were chosen as the crankshaft and camshaft position sensor signals. These signals were manipulated to allow software programmable engine speed selection in the HiL test system. Note that it is possible to supply the crankshaft signal and to supply an additional pressure or temperature signal. A camshaft signal error will be present in that case. It is also possible to use a PC card with more analog outputs or to use two or more PC cards with two analog outputs each to increase the number of engine sensors with analog outputs that can be simulated. None of these two approaches were pursued in the work reported here.

Engine ECU HiL simulation requires real time generation of the signals to be supplied to the ECU by the simulator. The software architecture of the HiL simulator presented here is illustrated in Figure 2. This software architecture is based on the Simulink®, Real Time Workshop® and Real Time Windows Target® modules of Matlab®. All simulation programming is done using the block diagram programming in Simulink®. This block diagram is automatically changed into C code by the Real Time Workshop®. This C code is linked with the code for the PC card's I/O function C codes, downloaded to the PC card and run in real time using Real Time Windows Target®. Real Time Windows Target® is a real time system that suppresses some capabilities of Windows and runs the real time code alongside other Windows applications. With this architecture, engine speed can be changed during the HiL simulation. It can also change in an a priori programmed fashion.

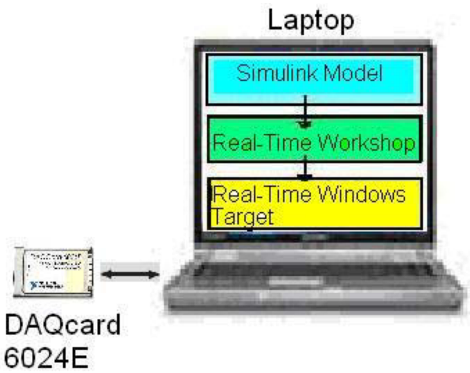

Figure 2. Hardware architecture of the HiL test system

IV. CRANKSHAFT AND CAMSHAFT POSITION SENSOR SIGNAL GENERATION

The engine ECU used in this study controls a six cylinder engine. The crankshaft position sensor generates 60 peaks per revolution meaning a resolution of 6 degrees/peak. For indexing purposes, two of these peaks are empty. So, alltogether there are 58 peaks and 2 null outputs in the crankshaft position sensor voltage as a function of crankshaft angle for one full revolution. The camshaft position sensor voltage output has 6 peaks corresponding to the six cylinders and one extra peak for indexing. Experimentally determined crankshaft and camshaft position sensor signals are shown in Figures 3 and 4, respectively.

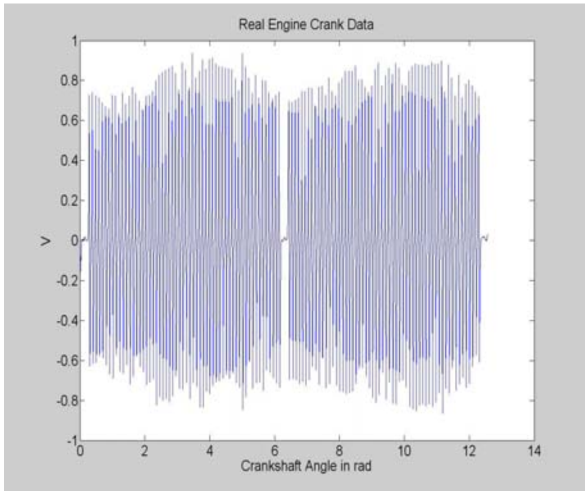

Figure 3. Experimental crankshaft position sensor signal

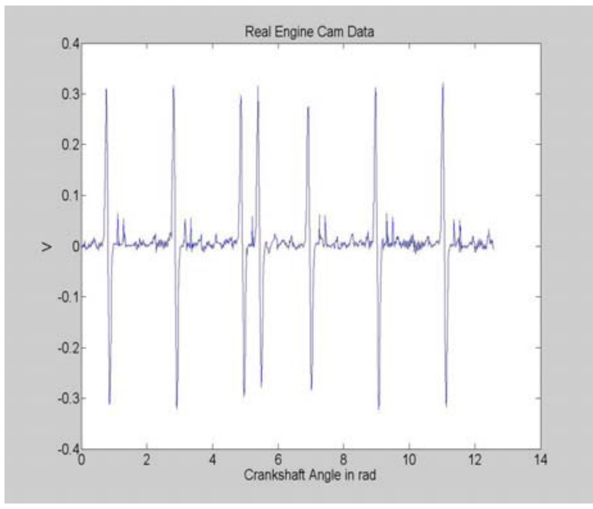

Figure 4. Experimental camshaft position sensor signal

Generation of accurate and synchronized crankshaft and camshaft position sensor signals is vital for correct HiL simulation. Both signals are first created offline as a function of crankshaft angle. The crankshaft and camshaft signals as functions of crankshaft angle are shown in Figures 5 and 6, respectively. The x axis in these two figures corresponds to two full crank revolutions. Note that the camshaft goes through one revolution for two revolutions of the crankshaft.

Note that crank and camshaft sensor waveforms are fixed (except for noise effects) as a function of crankshaft angle. These two waveforms are thus created and stored before the real time HiL simulation takes place. A separate program with a graphical user interface has been prepared to create these two waveforms offline, i.e. before the simulation. This program also allows the user to create crank and camshaft signals with a variety of faults. Possible faults which can be created by the program include missing peaks in the crank or camshaft sensor, changes in width or height of chosen parts of the signal (usually the peaks) and the addition of sensor noise. It is also possible to inject these faults while the real time HiL simulation is running.

V. HIL SIMULATION TEST RESULTS

The results of five different tests conducted using the HiL simulator will be presented here. The first two tests are variable engine speed tests. In the first two tests, the signals corresponding to the variable engine speed are generated in the Simulink® program and sent to the ECU. The third test uses a mean value engine model with throttle input for generation of the test signals. The fourth test investigates the effect of throttle position and engine speed changes on injection timing. The last test uses synthetically generated faulty signals to observe ECU operation in the presence of sensor faults. HiL simulation playback of recorded engine sensors during actual vehicle operation has also been carried out but is not reported here.

The main Simulink® simulation block diagram is shown in Figure 7.

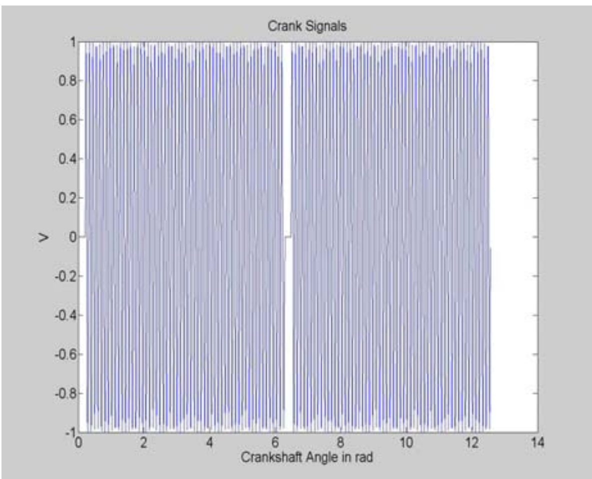

Figure 5. Crankshaft position sensor signal

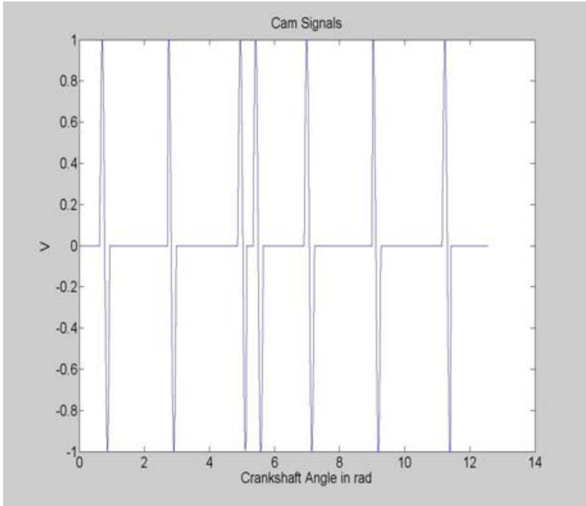

Figure 6. Camshaft position sensor signal

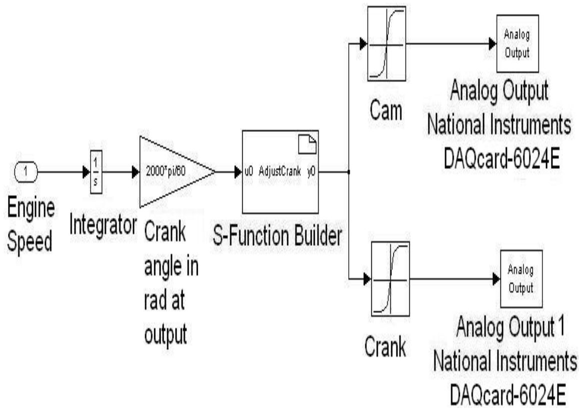

Figure 7. Main Simulink® simulation block diagram

A. *Engine Speed Ramp*

The first test is a very simple one, aiming to demonstrate the variable engine speed capability of the simulator. The HiL simulator generates crank and cam position sensor signals that correspond to a ramp type increase in engine speed. The ECU monitoring software INCA ran continuously in a PC during all the tests reported here. This software is used to verify correct operation and to monitor the ECU calculated engine speed in its soft oscilloscope screen. The results of the first three tests are presented using snapshots of these soft oscilloscope screens where the x axis is time in seconds and the y axis is the ECU computed engine speed in rpm. The ramp response in Figure 8 shows that the desired engine speed ramp is generated correctly.

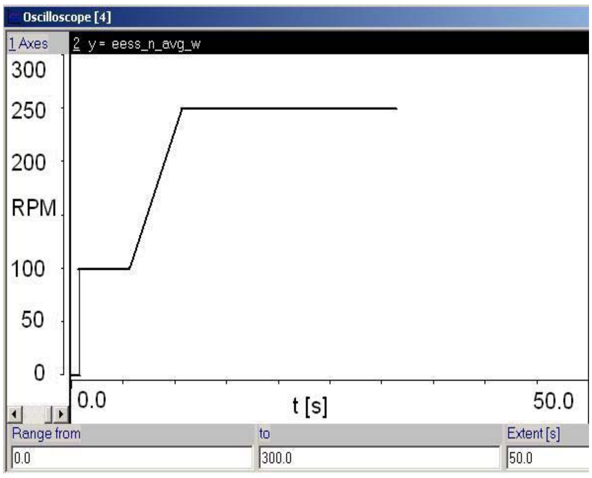

Figure 8. Engine speed ramp from ECU

B. Engine Speed Sine Wave

The crank and cam shaft position sensor signals are created to generate a sine wave type engine speed in this test. The frequency of the engine speed variation is 0.25 Hz. The sine wave response is shown in Figure 9.

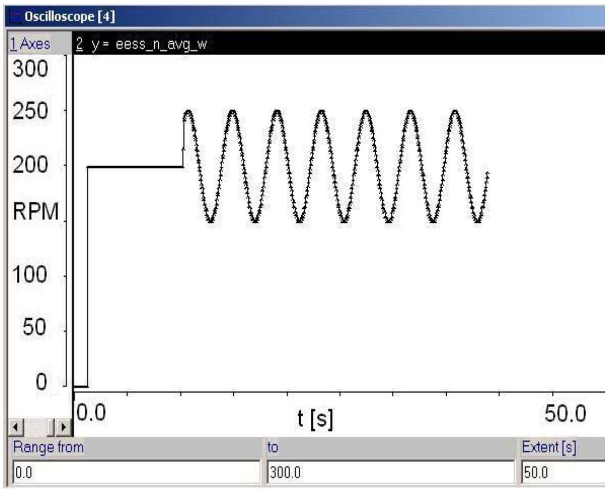

Figure 9. Engine speed sine wave from ECU

It is clear that the sine wave engine speed can be generated correctly by the HiL simulation test system.

Note that the engine speed signals used in the two tests presented above are not realistic. Real engines in real vehicles cannot generate such engine speeds. The next test, therefore, uses a mean value engine model for more relistic determination of engine speed.

C. MVEM Model Test

A simple diesel engine model that will be used for real time computation of engine speed and pressure and temeperatures of interest based on throttle and ECU injection signals is under preparation. As this model is not ready and as commercial diesel engine models are very expensive, full fledged engine ECU HiL operation could not be achieved. In order to make sure that the HiL system in this paper can run a simple engine model in real time operation, an available mean value engine model (MVEM) was used to simulate the dynamic computation delay that will exist when an engine model is incorporated into the system. An MVEM available in the literature [8] and coded in the engine blockset in the references [5-6] was, therefore, used in this test.

The MVEM model part of the Simulink® block diagram used is taken from the abovementioned references and is shown at the bottom of Figure 10 along with the signal generation part at the top of the same figure. The output of the MVEM model is engine speed and its input is throttle position. In the HiL simulation, a step change in throttle position is commanded to the engine model and the resulting engine speed change is sent to the engine ECU. The MVEM model response is shown in Figure 11.

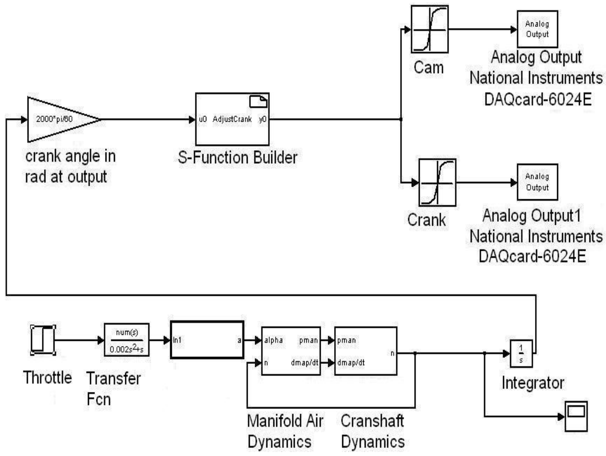

Figure 10. MVEM test Simulink model

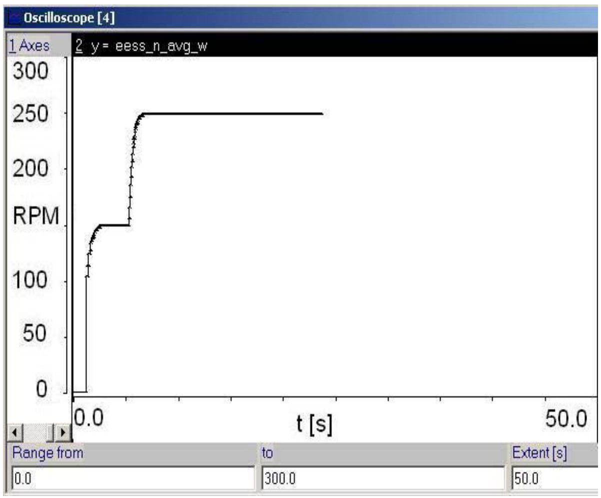

Figure 11. Engine speed in MVEM test from ECU

D. Injection Signals

Due to the unavailability of a diesel engine model, real ECU control tasks could not be tested. Instead, the throttle position sensor input to the ECU was changed and the injection signals generated by the ECU were captured and analyzed for different engine speed values. A typical injection signal for one of the cylinders is shown in Figure 12 at an engine speed of 700 rpm. Smaller throttle inputs and lower rpm values resulted in shorter injection signals as expected.

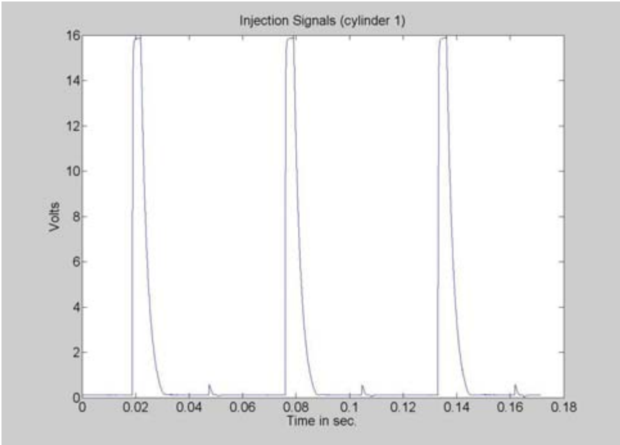

Figure 12. ECU injection signal at 700 rpm

E. *Faulty Signal Test*

A program with a graphical user interface was created in Matlab® for synthetic generation of faulty signals. Some typical faulty signals that can be generated with this program are shown in Figures 13 and 14. The signal in Figure 13 has two missing peaks corresponding to two revolutions of a crankshaft with one broken sensor tooth. The signal in Figure 14 has one broken camshaft sensor tooth. HiL simulation with these signals results in crank and cam errors being reported by the ECU. The program that was prepared can generate a large variety of faulty signals other than missing peaks in crank and cam signals.

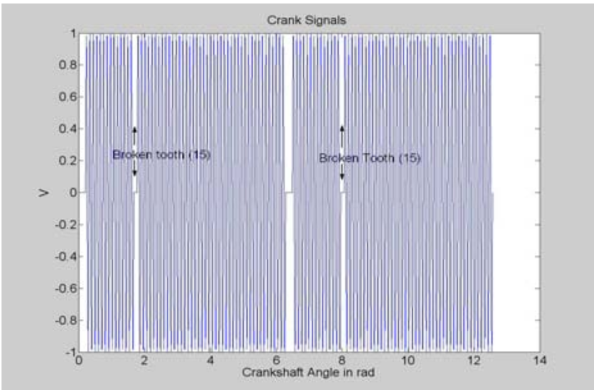

Figure 13. Crankshaft signal with missing peaks

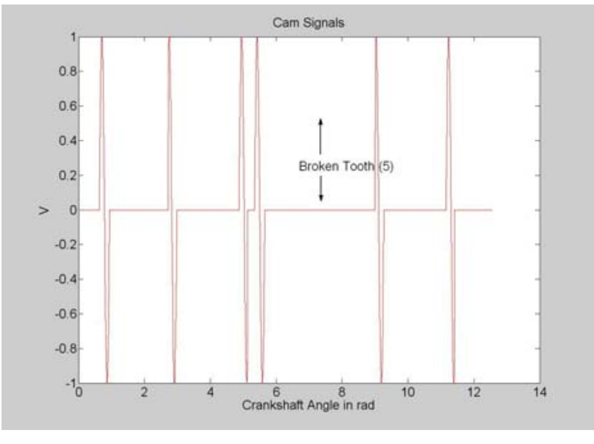

Figure 14. Camhaft signal with missing peaks

VI. CONCLUSIONS AND RECOMMENDATIONS

A low cost and portable engine ECU HiL test system was presented in this paper. The capability of simulating variable engine speed, using engine models in the simulation and generating and testing faulty signals was demonstrated using experimental results. Being a low cost solution, the system presented here does have some shortcomings like being limited in maximum engine speed and the number of channels used in HiL simulation. The maximum engine speed that can be simulated is based on the processor performance of the laptop and the real time operating system used. The average level performance laptop used here with an Intel Celeron 2.70 GHz processor and 256 Mbyte ram memory could simulate a maximum engine speed of 2000 rpm during the tests. The ECU used had a maximum defined engine speed of 5400 rpm. If a portability factor based on the ease of mobility of the system is defined, however, this system will rate highly as it occupies as much space as a laptop. Speed can be improved to a certain extent by the use of higher capability PC cards and a faster laptop.

Future work will concentrate on the incorporation of a diesel engine model into the HiL test system. Real ECU control tasks like idle speed control can be tested very easily in that case. The generation of combustion pulsations in the engine signals also seems worthy of investigation.

ACKNOWLEDGMENT

The first two authors would like to thank Ford Otosan for supporting the work presented in this paper through grant ÜG-03.004.

REFERENCES

[1] A. Kimura and I. Maeda, “Development of engine control system using real time simulator,” *Proceedings of the 1996 IEEE International Symposium on CACSD,* pp. 157-163, September 15-18.

[2] R. Isermann, J. Schaffnit and S. Sinsel, “Hardware-in-the-loop for the design and testing of engine-control systems,” *Control Engineering Practice*, vol. 7, 1999, pp. 643-653.

[3] J. Schaffnit, S. Sinsel and R. Isermann, “Hardware-in-the-loop simulation for the investigation of truck diesel injection systems,” *Proceedings of the American Control Conference. Philadelphia, Pennsylvania,* pp. 498-502, June 1998.

[4] H. Hanselmann, “Hardware-in-the-loop simulation testing and its integration into a CACSD toolset,” *The IEEE International Symposium on Computer-Aided Control System Design. Dearborn, Michigan,* 1996, September 15-18.

[5] B. Aksun Güvenç, B. Sencer, M. Giray and L. Güvenç, “Use of a Simulink engine blockset in real time hardware in the loop simulations,” *IEEE International Conference on Mechatronics, İstanbul,* 2004, June 3-5.

[6] B. Aksun Güvenç, B. Sencer, M. Giray and L. Güvenç, “A control oriented Matlab/Simulink IC engine library,” *(in Turkish), Otekon04 Automotive Technologies Congress, Bursa,* pp. 415-422, June 21-23.

[7] A. Çebi, Testing an Engine Electronic Control Unit in Real-Time Hardware-in-the-Loop Simulations, (in Turkish), M.S. Thesis, Istanbul Technical University, 2004.

[8] E. Hendricks, “The Analysis of Mean Value Engine Models,” SAE 920682, 1992.