6. ANKARA INTERNATIONAL AEROSPACE CONFERENCE
14-16 September 2011 - METU, Ankara TURKEY

AIAC-2011-085

SYSTEM IDENTIFICATION AND CONTROLLER DEVELOPMENT FOR A SMALL TURBOJET ENGINE

Sinan Ekinci[1]
Aerotim Ltd.
Ankara, TURKEY

Ali Türker Kutay[2], Oğuz Uzol[3]
Dept. of Aerospace Engineering,
Middle East Technical University
Ankara, TURKEY

Melih Yiğittürk[4]
TUSAS Engine Industries, Inc.
Eskişehir, TURKEY

ABSTRACT

This paper presents the results of a system identification and controller development study for a small generic turbojet engine. Custom Full Authority Digital Engine Control (FADEC) system hardware and embedded software are designed and implemented on the engine. The system also controls the automatic start-up sequence for the engine to ensure a safe and guaranteed ignition under all-weather conditions as well as safe acceleration until the engine reaches the idle speed. The shut-down sequence is also controlled to make sure the engine is stopped and cooled-down in a safe manner. Both of these sequences are operated in open-loop. For closed-loop feedback control operation, a dynamic mathematical model of the engine is obtained through a series of system identification tests. The developed FADEC system is used to supply the engine a pre-determined fuel flow rate schedule in open-loop. Various step signals are applied to the fuel pump and the engine's responses to these commands are recorded. Linear dynamic models are fitted to the responses to represent engine behavior at different speeds. The obtained models are then used to design feedback controllers to regulate engine speed despite changing loads and environmental conditions. Performance of the developed FADEC is demonstrated through bench tests.

INTRODUCTION

Operation of a gas turbine engine at maximum efficiency for a given condition requires precise control of engine parameters such as fuel flow, stator vane position, bleed valve position, etc. The complexity of this task is proportional to the complexity of the engine. Originally aircraft engine control systems consisted of simple mechanical linkages controlled by the pilot. As the performance of the engines improved and duties of the pilots got more complex, engine control systems evolved and required a flight engineer on board to operate the engine. After mechanical engine control systems, analog electronic engine control was introduced, where desired engine settings were changed by varying electrical signals. Digital control systems followed analog controllers.

Almost all gas turbine engines today are controlled by Full Authority Digital Engine Controllers (FADEC) with no form of manual override, giving full authority over engine settings to a computer. FADEC adjusts engine parameters to provide the desired power output as commanded by the throttle position while monitoring various engine variables. FADEC also controls engine starting and

---

[1] Aeronautical Engineer, Email: sinan.ekinci@aerotim.com.tr
[2] Assistant Professor, Email: kutay@metu.edu.tr
[3] Assistant Professor, Email: uzol@metu.edu.tr
[4] Control Design Engineer, Email: Melih.Yigitturk@tei.com.tr

restarting sequences. It is based on a computer with necessary peripherals to read various sensors and drive various actuators. It also has an interface to communicate with the user to receive commands and transmit information related to engine performance. Efficient operation of the engine requires processing of the collected data from the sensors to find the optimum engine settings. The algorithms for engine control are programmed into FADEC as embedded software.

Numerous approaches for closed-loop control of turbine engines have been proposed in the literature. Multivariable control of the GE T700 engine where the fuel flow rate and variable geometry inputs are varied simultaneously to control power turbine and gas generator speeds is presented in [1]. To compensate for high frequency uncertainty in the system model due to unmodeled rotor dynamics the LQR/LTR design methodology is employed. Instrumentation of a small engine with rotation, pressure, and temperature sensors and a computer for feedback control of engine speed is described in [2]. First a dynamic model is defined and identified based on test data and a PI controller is designed using the experimentally obtained model to control the rotational speed as measured in revolutions per minute, or RPM. Closed-loop tests showed improvements in the dynamic response of the system. Design of a 2-DOF PID controller for a gas turbine is explained in [3]. Gains of the controller are tuned using Adaptive Neural Network Fuzzy Interference System. In a similar work in [4] gains of a PID controller for a Heavy Duty Gas Turbine Plant are tuned using genetic algorithm. The resulting controller is shown to have optimal performance when compared with PID controllers designed using Ziegler-Nichols' method and performance index method. A state space controller was designed by LQR and pole placement methods using a 15-state model of a twin shaft gas turbine [5]. A high order linear model was obtained through linearization of a nonlinear model and MIMO state-space control methods were employed to avoid gain scheduled controllers. In [6] a MIMO multi-agent control system was designed for single-shaft heavy-duty gas turbine to control shaft rotational speed and stack temperature by using fuel mass flow and variable inlet guide vanes as control inputs. Effective cancellation of disturbances due to changes in load conditions was demonstrated.

The motivation for the current study originated from the need for developing necessary hardware architecture and software algorithms to be applied on indigenous FADEC systems developed by Aerotim Ltd. These systems are to be applied on the small turboprop and turbojet engines that are being developed by TUSAŞ Engine Industries (TEI) for various air vehicles. Therefore custom FADEC systems are required that can be fully optimized for every variant. The turboprop version of the TP38 engine is shown in Figure 1.

Figure 1: *TP38 small turboprop engine.*

In this paper we present the results of a system identification and controller development study for a small generic turbojet engine, the core of which is similar to the engines designed by TEI. Custom FADEC system hardware and embedded software are designed and implemented on the engine. The system also controls the automatic start-up sequence for the engine to ensure a safe and guaranteed ignition under all-weather conditions as well as safe acceleration until the engine reaches the idle rpm. The shut-down sequence is also controlled to make sure the engine is stopped and cooled-down in a safe manner. Both of these sequences are operated in open-loop. For closed-loop feedback control operation, development of an appropriate controller requires a verified dynamic mathematical model of the engine. This is obtained through a series of system identification tests. The developed FADEC system is used to supply the engine a pre-determined fuel flow rate schedule in open-loop. Various step signals are applied to the fuel pump and the engine's responses to these commands are recorded. Linear dynamic models are fitted to the responses to represent engine behavior at different speeds. The obtained models are then used to design feedback controllers to regulate engine speed despite changing loads and environmental conditions. Bench tests performed with the developed FADEC show that desired engine speeds externally given to the FADEC through a thrust lever are successfully achieved within less than ten seconds. The experience gained in this work will be used towards developing customized FADEC systems for new engines designed and developed by TEI.

TEST SETUP

In order to develop and test appropriate hardware and software components for the FADEC system, an existing small turbojet engine in the Aerospace Engineering Propulsion Laboratory (AEPL) of METU is used as a testbed. This engine is mainly a commercially available simple small turbojet that is designed for radio-controlled modeling applications. It has a single stage radial compressor driven by a single stage axial turbine and a through-flow combustor. It uses JP8 jet fuel during normal operation but the start-up sequence involves a pre-heat mode using propane gas. The maximum rpm is around 120 000 at which it can produce about 100 N of thrust. The engine pack comes with a pre-installed FADEC system that controls the electric starter motor, the fuel pump, the glow-plug and two solenoid valves for JP8 and propane fuel lines and reads measurement signals from a magnetic RPM sensor and a K-type thermocouple that measures the Exhaust Gas Temperature (EGT). Figure 2 shows a view of the test setup.

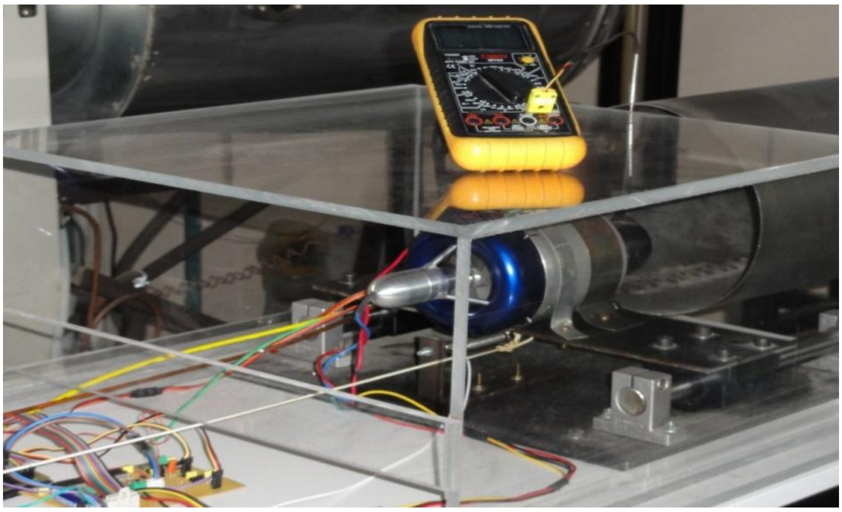

Figure 2: Small turbojet test setup

HARDWARE AND SOFTWARE DEVELOPMENT

Before starting the development of hardware and software components, first the jet engine is tested using its original FADEC system. During these tests general signal types and levels are measured and recorded for the system actuators (start-up motor, fuel pump, solenoid valves, glow-plug) and sensors (RPM sensor and EGT thermocouple) while the engine is operating in closed-loop. In addition, the characterization of the fuel pump is performed at this stage to determine the fuel-flow rate variation with applied voltage. These tests are performed on an isolated fuel-pump meaning that the outlet pressure level of the pump is atmospheric. Figure 3 shows the results of the fuel pump characterization tests.

As is evident, the pump operates linearly in a wide range of flow rates except for very low voltages where the pumping very quickly stops below 0.5 g/s. In the normal operation of the jet engine at idle and above idle RPMs this characteristic does not create any problems, however, it creates significant issues during the start-up sequence such that the fuel flow can very quickly jump from 0 to 0.5 g/s and this may result in a sudden rich burning within the combustion chamber which is usually observed as an extending flame from the nozzle. This can cause the EGT to exceed the thermal shut down threshold value of 850 ºC of the tested engine and may lead to engine failure. In order to start the engine smoothly without flame extension, fuel supply to the engine needs to be adjusted precisely in the dead zone region below 0.5 gr/sec. This is achieved by applying an on/off PWM signal to the fuel pump at 10Hz to drive fuel pump in its dead zone. At “On” position, the duty cycle of the PWM is set above the start level of the fuel pump and at “Off” position it is set to zero. By adjusting the duty cycle of the PWMs a smooth fuel ramp is obtained.

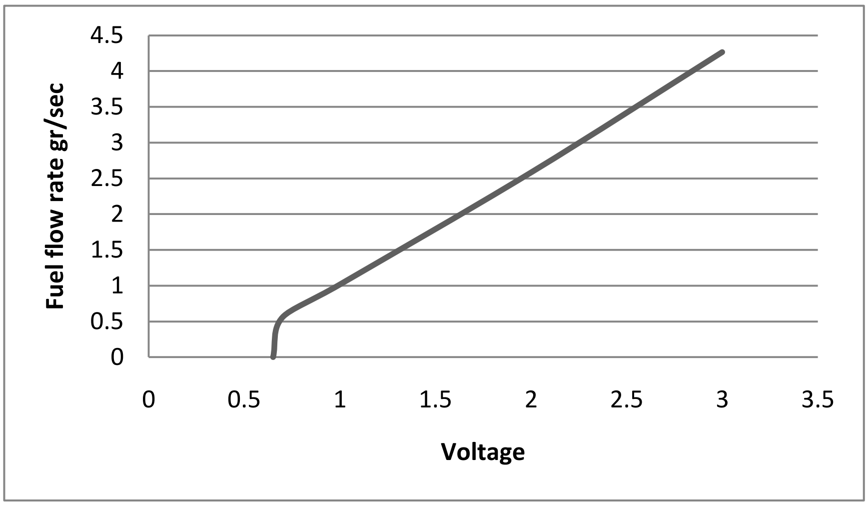

Figure 3: Fuel pump flow rate (g/s) vs. applied voltage

After these initial tests, the pre-installed FADEC of the engine is taken out of the loop and a prototype hardware is designed to replace the existing system. Appropriate drivers and electronic components are selected and a Printed Circuit Board (PCB) was designed to run the two DC motors (starter and the fuel pump), two on/off solenoid valves for JP8 and propane fuel lines and one low impedance resistor (the glow plug). Also appropriate electronic components are selected to read the RPM sensor and thermocouple outputs. To validate the prototype hardware, similar RPM and K type sensors are mounted on the engine. Original FADEC can store the last 60 minutes of engine data at a period of 0.5 seconds in its memory and by using the FADEC software these data can be downloaded as a txt file. The downloaded original FADEC data and the received data from the prototype hardware are compared in Figure 4, where “RPM measured” and “Temp measured” refer to data recorded using the prototype hardware. As seen in the figure, RPM sensor outputs are practically the same at the beginning of the test but later (for time ~2450 sec) a time lag is occurred between the FADEC data and measured data. This is because the original FADEC does not record data in real time, it records every time interval of the data with a time shift so as seen on the results the time lag is increasing during the test, because the time shift errors are accumulating. The same time lag can be seen from the temperature measurement. The measured EGT temperature and FADEC temperature have different transient characteristics, but they converge at steady state. The differences are mainly because of the slight shift in the axial and radial locations of the two thermocouples.

One of the most critical parts of the study is to develop and implement a robust and reliable start-up sequence for the engine that will ensure a safe ignition and acceleration up to the idle rpm under all-weather conditions. The engine start-up is very sensitive to the environment temperature and during very cold days the ignition may not be achieved. This was also an issue with the original FADEC system, and to overcome this problem a modified start-up algorithm is designed, implemented and tested on the jet engine. This algorithm begins with a user interrupt, which makes the starter motor run the compressor. After the propane line valve turns on, the glow plug ignites the propane. If the EGT exceeds its threshold level of 100 ºC the ignition flag is set to high and algorithm continues running, otherwise the propane valve turns off and starter stops. After successful propane ignition, JP8 valve is turned on and the fuel pump starts to pump fuel slowly into the engine. This phase is the most critical part of the startup algorithm because both types of fuel (propane and JP8) are being sent to the combustion chamber. When JP8 starts to burn, The EGT increases suddenly and approaches to the engine EGT threshold. At this instant the propane valve is turned off to protect the engine and then fuel ramp starts to push the engine towards its idle RPM. During startup the starter motor adjusts the rotational speed of the turbine shaft to its threshold level of 15,000 RPM. If the shaft RPM exceeds this threshold value, starter motor stops and its mechanical connection with the shaft is severed by the clutch automatically. This algorithm is coded using C programming language and embedded in to a PIC microcontroller on the prototype board.

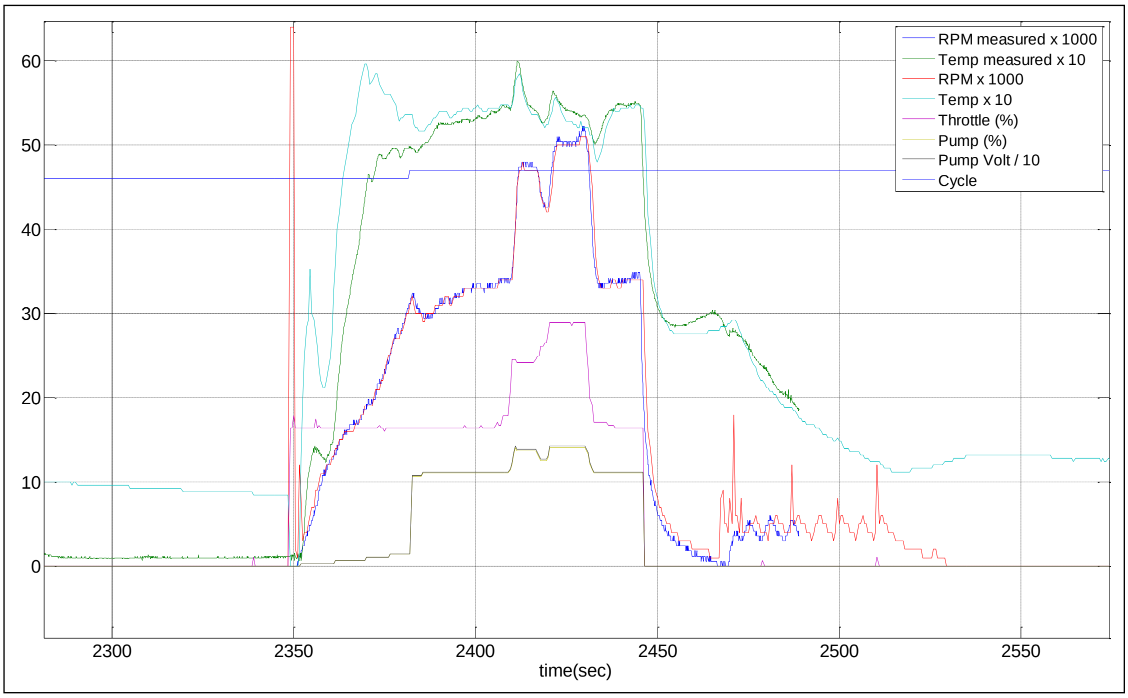

Figure 4: Prototype FADEC hardware validation test results

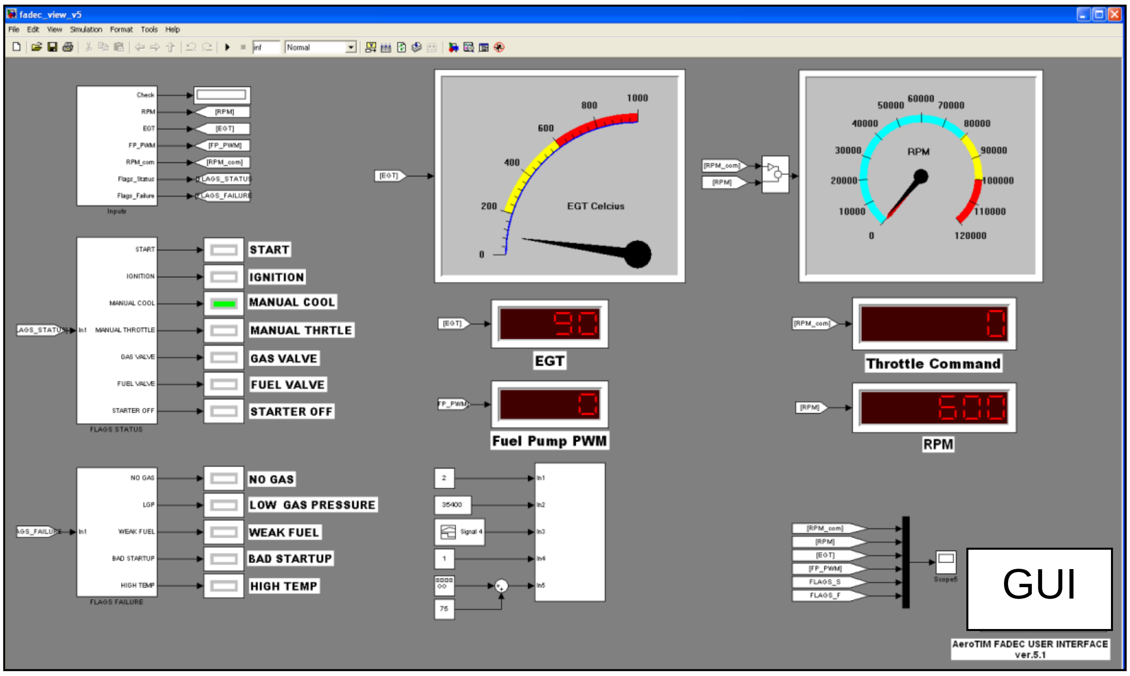

Figure 5: GUI designed to monitor the operation of the engine using the prototype FADEC

In order to monitor the measured signals from the sensors as well as to monitor the generated control commands during the operation of the engine a Graphical User Interface (GUI) is designed. Figure 5 shows a screenshot of the designed GUI. The interface displays real-time information for the measured EGT and RPM signals as well as the PWM signal that is being fed to drive the fuel-pump. A collection of LED signals is used to alert the user for different conditions during the operation of the engine.

SYSTEM IDENTIFICATION and MATHEMATICAL MODEL DEVELOPMENT

For closed-loop feedback control operation, development of an appropriate controller requires a verified dynamic mathematical model of the engine. In this study, this is obtained through a series of system identification tests. The developed prototype FADEC system is used to operate the engine in open-loop and to supply the engine pre-determined fuel flow rate schedules. Various step signals are applied to the fuel pump and the engine's responses to these commands are recorded. These tests are repeated multiple times to check repeatability and to obtain an average response of the system. Figure 6 shows a sample result showing the RPM response of the engine to a series of step inputs to the fuel-pump.

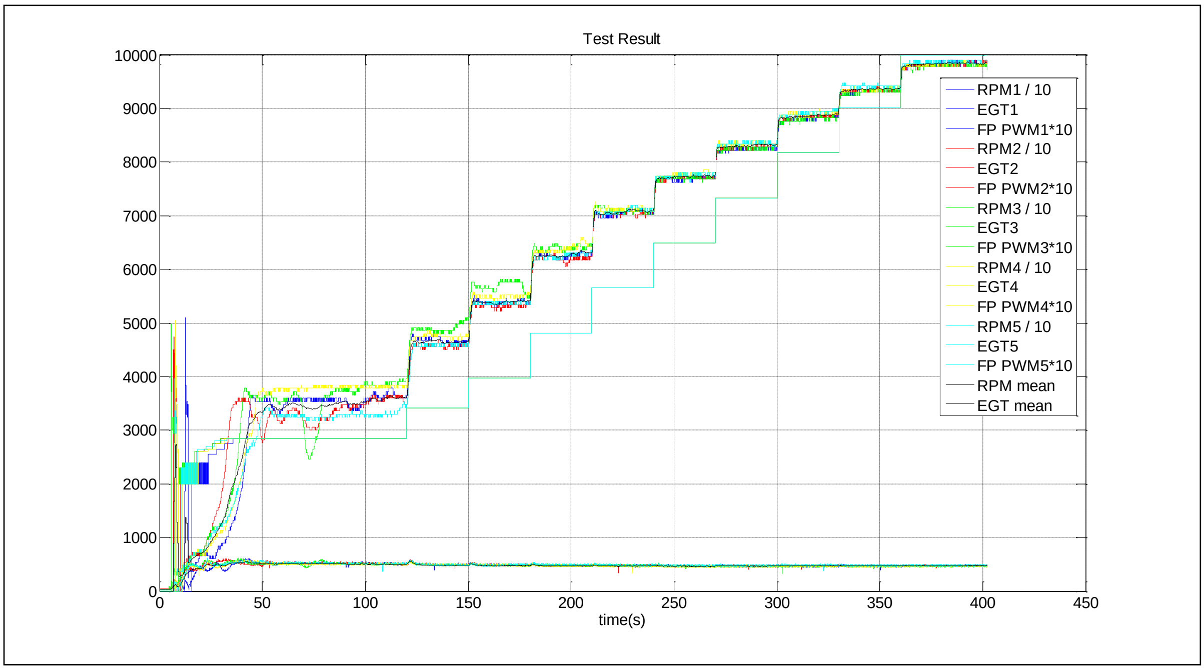

Figure 6: System ID test results

As seen in Table 1, a second order linear dynamic system represented by a gain, natural frequency, and damping ratio is fitted to each step response obtained at various RPMs. A second order system and its parameters can be defined as follows;

$$G(s) = \frac{K.\,\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

The obtained set of linear models is then connected to form a nonlinear dynamic model for the entire speed range. The resulting model is a second order system whose parameters depend on RPM. For every RPM value the system parameters to be used are obtained through interpolation from the parameters of the linear models fitted for a number of RPMs.

Table 1: Second Order System Parameters for Linearization RPMs

	Linearization RPMs	**K**	ζ	ω_n
System1	36000	126	0.70	1.6
System2	46500	137	0.70	1.4
System3	54000	136	0.90	1.5
System4	63000	129	0.80	2.0
System5	70000	125	0.90	1.7
System6	77000	118	0.95	1.8
System7	83000	113	0.99	1.8
System8	89000	108	0.99	2.0
System9	93500	103	0.99	2.0

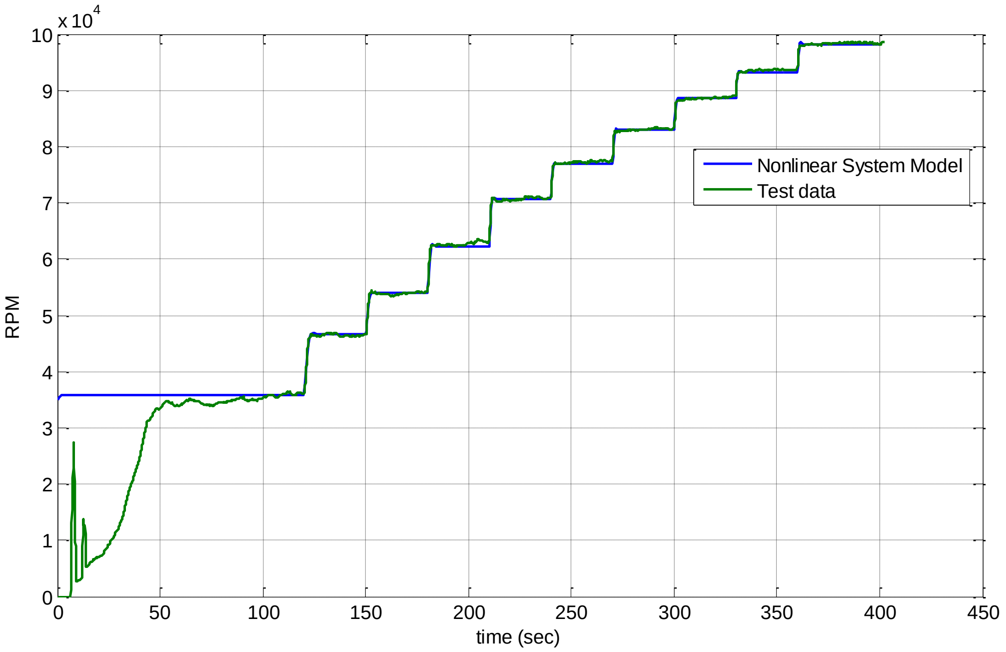

Figure 7: Comparison of the linear system response of the produced mathematical model with the test results

Figure 7 shows the response of the nonlinear model to a series of step inputs in fuel flow and the result is compared to the test data. As is evident the produced mathematical model is able to simulate the engine behavior quite well and it will be used to design feedback controllers to be implemented on the engine.

CONTROLLER DESIGN

For controller design process, linear system models obtained from system identification tests are used. To obtain lowest possible steady-state error and overshoot rates, PI controller structure is preferred. As a design constraint maximum overshoot is limited to 1%.

As seen in Table 1 the highest natural frequency observed in the entire operation envelope is $\boldsymbol{\omega_n}$ = 2 rad/sec (0.32 Hz). The developed FADEC runs at 10 Hz so its frequency is 30 times faster than the engine dynamics and this ratio is quite enough to control the engine in the entire operation range. The objective of the controller is to remove steady state error and provide uniform engine response throughout the entire speed range by setting the closed loop system frequency to approximately 2 rad/sec. For every linear model identified a PI controller is designed using the root locus method [7]. The resulting controllers are scheduled with RPM to achieve uniform closed loop performance at every speed [8].

The transfer function of a PI controller can be written as

$$\begin{aligned} G_{PI}(s) &= K + \frac{K_I}{s} \\ &= K\left(\frac{s+a}{s}\right). \end{aligned}$$

The zero of the controllers $a = K_I/K$ is placed -3 rad/sec. For this zero location the root-locus of the closed loop system is shown in Figure 8 for System 1. The gain of the controller is set to 0.006 to obtain minimum rise time and maximum overshoot ratio. The step response of the closed loop system is shown in Figure 9.

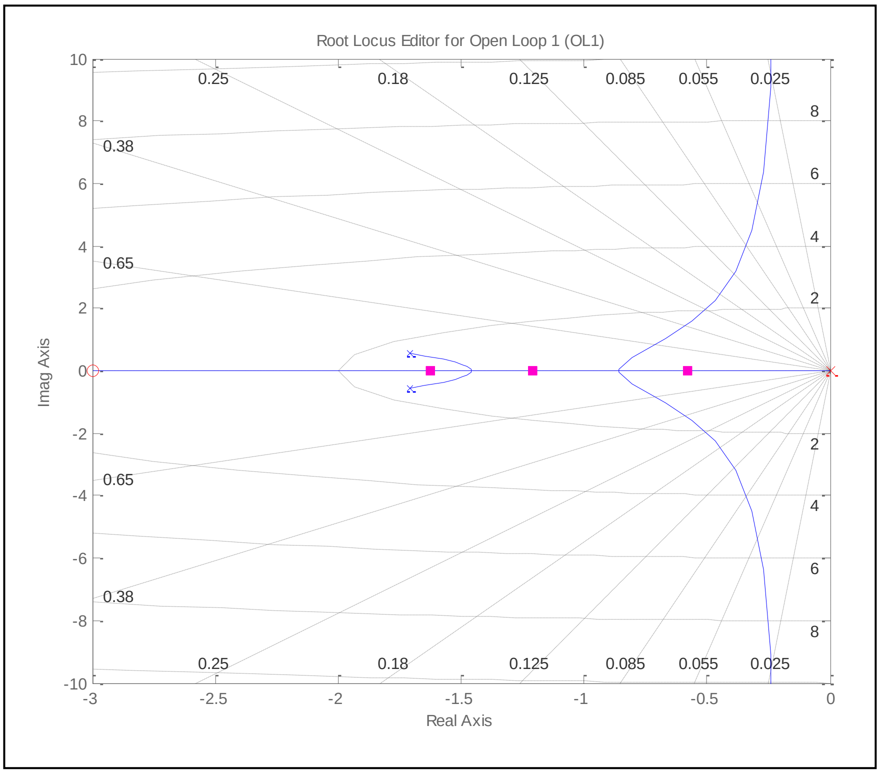

Figure 8: Root-Locus for System

Figure 9: Step response of the closed loop system for gain: 0.006

For the other linear systems, place of the zero is kept at same location (-3 rad/s) and the gain is adjusted to obtain best response in controller design limits. Finally, nine different sets of controller parameters are obtained for nine linear systems. For every RPM value the controller parameters to be used are obtained through interpolation from these parameters.

CONTROLLER VALIDATION

The designed controller structure and MATLAB Simulink model is shown in Figure 10. In Figure 10 the block Jet Engine contains the nonlinear engine model explained previously. This Simulink model is used to test the designed controller. The controller is also coded and embedded into FADEC and evaluated on the test engine by commanding various desired engine RPM values through a series of steps. The real test RPM measurement and RPM command is compared to the simulation results to validate the processes of discretization and embedding of the controller in microcontroller.

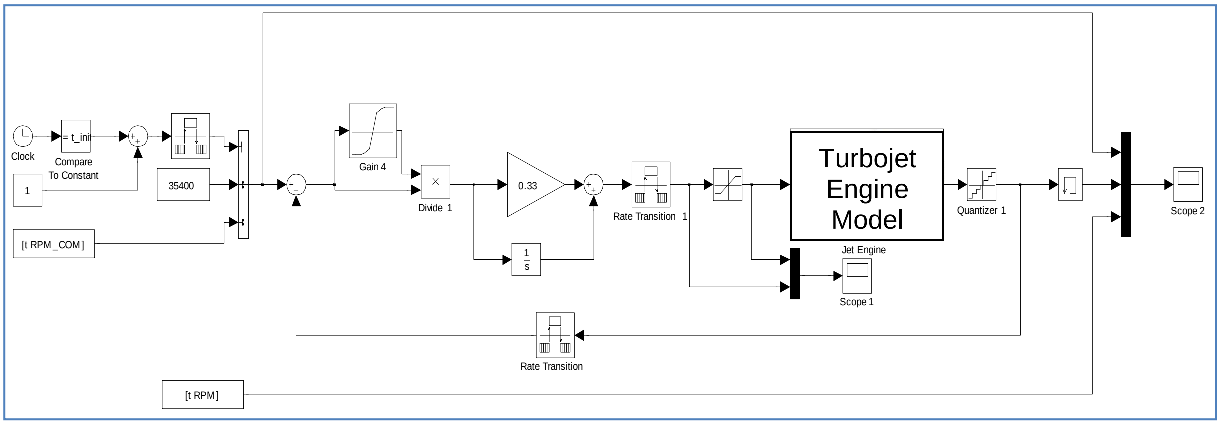

Figure 10: Closed Loop Test model

In the first test, the engine RPM is increased to 55,000 with small steps and then decreased back to the idle RPM of 35,400. During the test, the controller works as expected and the engine follows the commanded RPM values. Then for validation, the same RPM command profile is fed to MATLAB Simulink model.

The RPM sensor creates one high level signal for one engine shaft revolution and FADEC counts these signals for 100 milliseconds intervals, so the count values should be multiplied by the scale factor of 600 to be converted RPM. This scale factor defines the RPM sensor quantization level, so small changes at engine shaft rotational speed in test measurements creates spikes of 600 RPM. These measurement errors are called quantization errors. Despite the quantization errors, as seen in Figure 11 and Figure 12, simulation and test results are very close to each other.

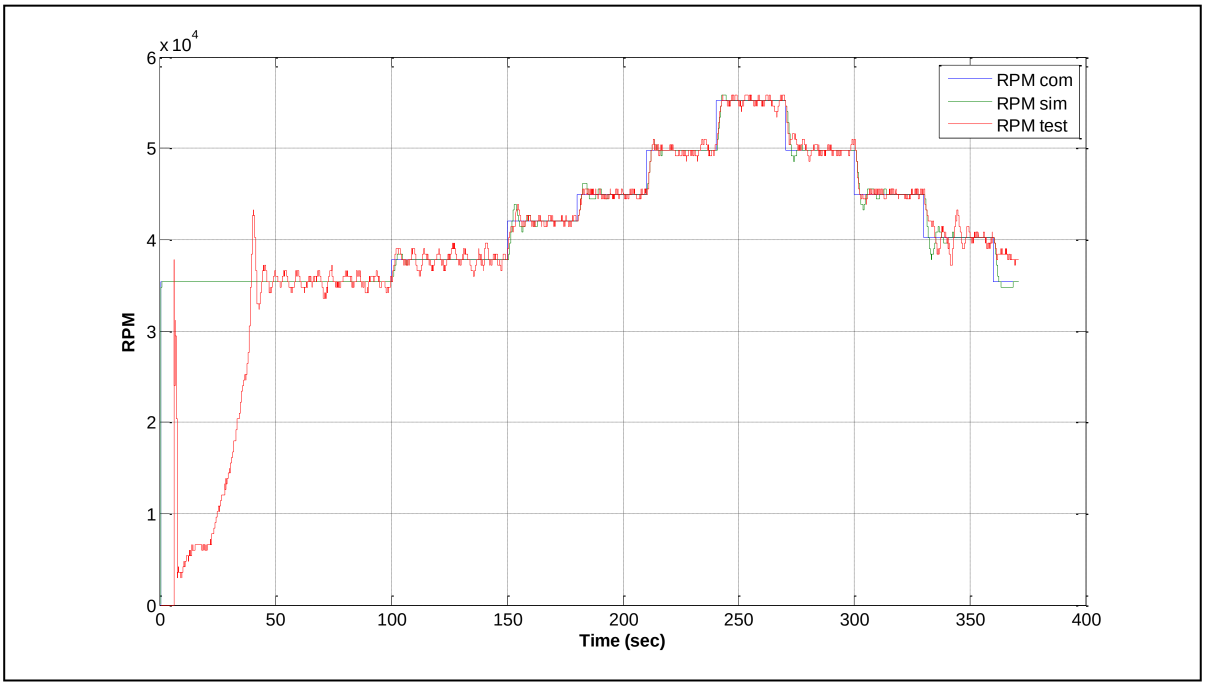

Figure 11: Closed Loop Controller test and simulation results Test-1

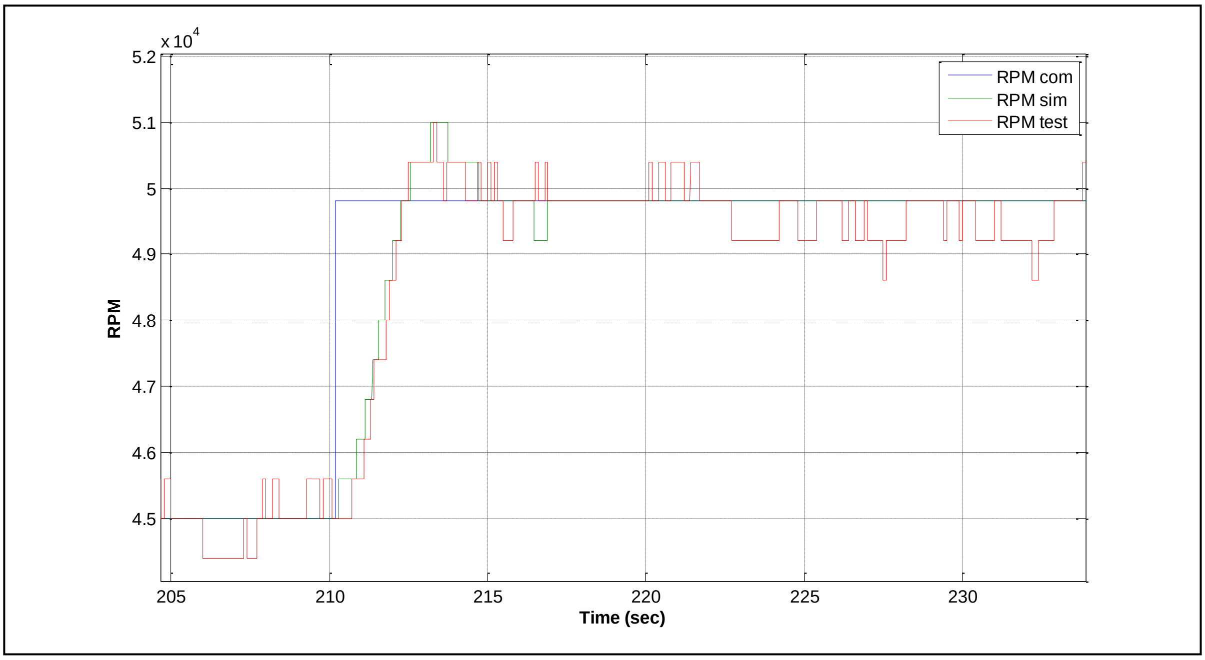

Figure 12: Comparison of closed loop test and simulation results Test-1 (Close view)

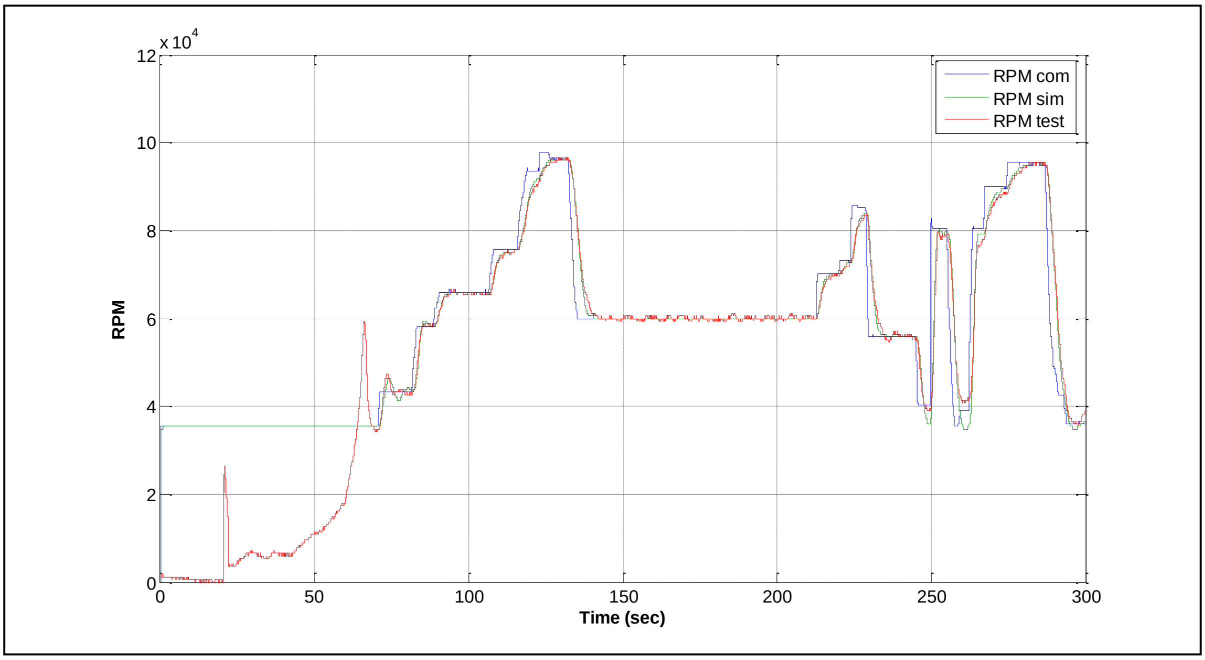

Figure 13: Closed loop controller test and simulation results Test-2

In Figure 13, The random RPM commands in full operational range is sent to the FADEC and its response is compared with the simulation results. As is evident, engine RPM follows the commanded values and the simulation data are merged with the real test results.

CONCLUSIONS

In this paper, the results of a system identification and controller development study for a small generic turbojet engine are presented. Custom designed hardware and embedded software are used for the development of a Full Authority Digital Engine Controller (FADEC) to replace an existing commercial controller. Open-loop tests are performed with various step fuel inputs to the engine and the responses are recorded, which in turn are used to derive a mathematical model of the engine. This model is then used to design a controller to be applied on to the engine. The designed controller is embedded in FADEC hardware and then tested. For validation of the design processes, the test and simulation results are compared with each other. As a result, engine RPM follows the RPM commands and the simulation results are merged with the real test measurements.

ACKNOWLEDGMENTS
We would like to thank TEI for the opportunity and the support given throughout this project.

References

[1] William H. Pfeil, Michael Athans, H. Austin Spang, III, *Multi-Variable Control of the GE T700 Engine Using the LQG/LTR Design Methodology*, American Control Conference, LIDS-P-1547, Seattle, WA, 1986,

[2] Thatiana Virgínia Granja Cruz, Janaína Gomes de Merícia, Carlos Gurgel Veras, Geovany Araújo Borges, *Instrumentation Architecture and Real-Time Control of Microturbine*, ABCM Symposium in Mechatronics, Vol. 2, pp. 35-42.

[3] Dong Hwa Kim, Chang Kee Jung, *New 2-DOF PID Controller Tuning by Adaptive Neural Fuzzy Inference System for Gas Turbine Control System*, Journal of Advance Computational Intelligence, Vol. 4, No:5, 2000.

[4] S.Balamurugan, R.Joseph Xavier, A.Ebenezer, Jeyakumar, *Application of Genetic Algorithm in Optimal PID Gain Tuning for Heavy Duty Gas Turbine Plant*, Journal of Electrical Systems, Vol. 4, No. 4. 2008.

[5] Markus Beukenberg, Michael Brodmann, Hans Weibel, Detlef Müller, *Design of State Space Controllers for Industrial Twin Shaft Gas Turbines*, Proceedings of GT2005 ASME Turbo Expo 2005: Power for Land, Sea and Air June 6-9, 2005, Reno-Tahoe, Nevada, USA

[6] Lorenzo Dambrosio, Marco Mastrovito, Sergio M. Camporeale, *Performance of Gas Turbine Power Plants Controlled by the Multiagent Scheme*, Transactions of the ASME, Vol. 129, pp 738-745, July 2007

[7] Ogata Katsuhiko, *Modern Control Engineering,Third Edition, Prentice - Hall*

[8] K. J. Astrom and B. Wittenmark, Adaptive Control, Second Edition, Addison-Wesley, 1995.