An Indirect Sensing Technique for Diesel Fuel Quantity Control

Final Report

ERIP Project 583
Assistance Instrument
DE-FG01-94CE15583

Prepared for the US Department of Energy, Office of Industrial Technology
David Crouch, Project Monitor
by C. Arthur MacCarley, Project Director
Loragen, Transportation Electronics Group
3576 Empleo Street, Unit 1, San Luis Obispo, California  93401
Document No. L-DOE-1-0599
Submitted:  May 25, 1999
# Table of Contents

Executive Summary ................................................................................................................................. 1  
Keywords .................................................................................................................................................. 1  
Disclaimer ............................................................................................................................................... 1  
Acknowledgements ................................................................................................................................. 1  
Project Accomplishments and Lesson Learned....................................................................................... 2  
Technical Tasks and Milestones ............................................................................................................. 8  
  Development of a practical, reliable real-time exhaust smoke sensor .................................................. 8  
  Development of a high-torque, high-speed throttle limit actuator ......................................................... 9  
  Computer simulation to aid in adaptive algorithm development ............................................................ 10  
  PC-based control development system .................................................................................................... 11  
  Microprocessor-based production prototype system ............................................................................... 11  
  Combined smoke control and mid-range particulate reduction studies ................................................ 15  
Actual Project Schedule .......................................................................................................................... 16  
Details of Hardware and Software Development .................................................................................... 17  
  Preliminary System Design and Specification of Software Development Tools ...................................... 17  
  Throttle Limit Actuator ............................................................................................................................ 18  
  Speed, Pressure, Temperature, and Throttle Limit Sensors ................................................................... 19  
  Exhaust Smoke Sensor ............................................................................................................................ 19  
  Control and Interface Software .............................................................................................................. 24  
  From Development System to Production Prototype .............................................................................. 26  
  System Power .......................................................................................................................................... 26  
  Production Sensors ................................................................................................................................. 27  
  Power Supply and Instrument Interface ................................................................................................... 31  
  Microcontroller Utilization ....................................................................................................................... 32  
  Control Code Development ...................................................................................................................... 32  
  Operator Interface .................................................................................................................................... 33  
Conclusions and Direction for Future Work ............................................................................................... 39  
Bibliography .............................................................................................................................................. 40  
Appendix .................................................................................................................................................... 42  
  Appendix 1. Selected Slides from Poster for DOE/OIT Exhibition, December 1998 ............................... 43  
  Appendix 2. DIESIM Simulation Program for EDSRS Development .................................................... 47
Final Project Report

An Indirect Sensing Technique for Diesel Fuel Quantity Control
ERIP Project 583
Assistance Instrument Number DE-FG01-94CE15583

Executive Summary

Financial support was provided by the Dept. of Energy Office of Industrial Technologies, via the Energy Related Inventions Program, for the development and commercialization of a method for improved diesel engine fuel quantity control with the objective of reducing exhaust smoke under full-load conditions. The proposed method involves advanced sensing, signal processing, and adaptive feedback control techniques. This report describes the accomplishments and results of this project. While originally conceived as a method suitable for OEM implementation only, modifications were made to utilize the technology in a retrofit system, applicable to existing diesel-powered vehicles. This required the development of an additional sensor and improved adaptive control algorithms, and verification testing on naturally aspirated and turbocharged diesel engines. The result of this work is a low-cost microprocessor-based control system that can be retrofitted to a wide range of diesel engines to prevent visible particulate emissions and poor efficiency caused by full-load overfueling. The system is designated the Electronic Diesel Smoke Reduction System (EDSRS). Dynamometer emission tests have confirmed the effectiveness of the system at maintaining exhaust smoke at 5% maximum opacity for all speeds. At the present state of development, a commercial prototype system is beginning in-service testing on diesel vehicles having a range of displacements from 1.8 L (naturally aspirated) to 7.3 L (turbocharged).

Keywords


Disclaimer

The contents of this report reflect the views of the authors who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the US Department of Energy or the Office of Industrial Technology. This report does not constitute a standard, specification or regulation.

Acknowledgements

We wish to thank the US Department of Energy, Energy Related Inventions Program for the support of this project. We wish to recognize David Crouch and Fred Hart, who served as monitors for this project, for their assistance, advice and patience over the long course of this project. This work is a direct extension of the proof-of-concept project funded 1989-91 by the National Science Foundation. The preliminary support of Isuzu Motor CO., Ltd is also recognized, including the donation of the test engine and COVEC electronic injection development system.

Technical work on this project was performed by C. Arthur MacCarley, Brian Hemme, Jim Gerhart, and Kyle Wayman, all of Loragen Advanced Technology Group. This report was written by the project director, C. Arthur MacCarley, with significant input from the senior project engineer, Brian Hemme. Except if noted, all figures, photographs and diagrams were created by the authors.
Project Accomplishments and Lesson Learned

The initial application for project funding was submitted to the DOE Energy Related Inventions Program in 1990. After an extensive review process, including analysis by two consultants and submission of additional data upon request, the project was funded in late 1994. Due primarily to competing academic commitments of the project director, an engineering professor at Cal Poly, San Luis Obispo, no-cost time extension requests were submitted to extend the period of performance ultimately to June 30, 1999.

The original concept for this project evolved from the experience of the project director in automotive alternative and diesel fuel engine research between 1982 and 1988. The original concept and system hardware description were published in 1988 in the SAE transactions. Proof of concept work was funded 1989-90 by the National Science Foundation. The results of preliminary development work were published in 1990 in the SAE Transactions. Support was provided by Isuzu Motor Co., Ltd. 1990-92 for further investigation of the sensing and control concept.

Active work on the project began in November 1996, with the establishment of the R&D company Loragen Systems for the purpose of commercializing this and similar automotive electronics products. A Federal EIN number (77-0445889) was obtained and DBA fictitious name statement was published for Loragen Systems in December 1996. Two engineering employees were hired in January 1997, and the services of a commercial payroll and business taxation firm were engaged.

Diesel engine technology changed considerable during the period between our grant application in 1990 and the initiation of work in 1997. New engines by most manufacturers now incorporate electronic controls, although these (as of the date of this report) remain open-loop with respect to either the actual fuel quantity or the exhaust emissions. The thrust of the industry is toward open-chamber, high-pressure direct injection (DI) engines, which have the potential for intrinsically lower emissions compared with conventional indirect injection (IDI) engines. Open chamber DI engines were previously considered impractical in small displacements, due to the resistance of atomized diesel fuel to compression-ignite outside of a protected precombustion chamber. The key to successful ignition and flame front propagation in a DI engine is higher injection pressure. Most diesel injection equipment manufacturers either have or are currently developing high-pressure pumps and nozzles for open-chamber DI applications. We modified a number of newer injection nozzles for high pressure direct injection applications. We experimented with and adapted our previous static and dynamic flow modeling algorithms for higher injection pressures for the modified DI nozzles.

An updated market assessment was performed, and a decision was eventually made, as advised by a DOE-recommended automotive marketing consultant, to reposition the target product as a retrofit system rather than an OEM licensed product. This decision recognized that (1) the automotive industry rarely, without a major legal confrontation, utilizes externally developed technologies, and (2) due to our small capitalization, it would be much easier to bring an emission-reduction/fuel economy improvement product to market through after-market equipment sales, possibly in conjunction with an established supplier of automotive performance equipment. This technical redirection required that we redesign the system to work with existing diesel engines and fuel injection systems rather than incorporating the proprietary indirect sensing method in the design of new equipment, which could only be done by an OEM equipment manufacturer. A number of new technical challenges were encountered in the areas of sensors and actuators that would work on a reasonable cross-section of the vast array of different types of diesel engines and injection systems. Much of the

development work supported by the ERIP grant was spent developing and refining these "general-purpose" components, and testing the system and components using both naturally aspirated and turbocharged engine configurations.

There are several key limitations in addressing retrofit applications rather than OEM "engineered-in" applications. The first limitation has to do with the injectors (injection nozzles). The initially-proposed indirect sensing method requires that one of the injectors be replaced with an equivalent unit instrumented with a needle lift sensor. On older engines, the replacement of one injector usually necessitates the replacement of all injectors, at considerable cost. Also, since injectors differ greatly between engines, even those using the same injection pump, the instrumentation and flow calibration of each requires custom engineering. The second key limitation is the requirement of an external means for limiting fuel delivery. On newer electronically controlled injection pumps, this can be accomplished by intercepting and modifying selected signals, as we have done with our Isuzu I-Tec test engine. On older diesel engines lacking any control electronics, an external actuator must be installed in a manner which allows it to override or modify the motion of the fuel sleeve or fuel rack actuator in the governor mechanism. While appears to be do-able for all the engines mentioned above, each installation and actuator is slightly different, and must be custom engineered for each engine series. The final limitation is that the proposed method requires knowledge of the exact smoke limit fuel quantity for the given engine, as a function of the commanded fuel delivery value (either a signal or actuator position) and the engine speed. This is best determined on a dynamometer engine test stand, since this measurement must be made at maximum engine load. An alternative, however, is possible, using a vehicle-installed external sensing means for smoke (high particle concentrations), in conjunction with "learning" algorithms to determine and store a smoke limit map under actual road conditions. We elected to focus on this alternative sensing approach, since it is potentially easy to deploy as a retrofit device and could be very effective is sufficiently reliable.

The retrofit embodiment we have developed adopts the last-mentioned solution for learning and implementing a smoke limit map. The retrofit package includes a means for directly sensing particulate output levels. Three methods were investigated: (1) opacity via long-path absorption or side-scatter, (2) ion transport whereby a small current flows between a high-voltage anode and cathode in the exhaust stream due to particulates serving as charge carriers, and (3) microwave scatter or reflection, roughly equivalent to a small radar system in the exhaust. Each real-time particulate method has been previously studied to various degrees by prior researchers or inventors, although integration in a closed-loop fuel control system of the type we are developing appears to be novel. Following extensive testing, were developed two types of sensors, full flow and sampled flow), both based upon exhaust gas opacity. This was found to be the most reliable and lowest in cost. The sensor is not novel in its basic sensing phenomena, which are well understood and used in other applications, but rather in the way that we can use these phenomena reliably in the exhaust stream of a diesel engine. The main problems of sensing particulates in the exhaust stream are the coking of the sensor elements (either optical or electrostatic) and the excessive heat in this environment. The sensor is unique in that it imbeds the sensor elements at each end a sensor section in "pockets" into which a small pressurized air stream is injected. This small air flow prevents intrusion of exhaust particulates onto the sensor elements and effectively cools the sensor elements, preventing damage. The air supply is provided by either a small bleed from a truck air brake system or a small electrically driven compressor. The sensor section is located in the exhaust plumbing between the exhaust manifold and muffler. A sample version of the particulate sensor, based upon the same method of operation, has also been developed for applications in which the installation of a sensor section in the exhaust plumbing is infeasible.

Two types of electromechanical actuators were considered, stepper motor driven and servomotor driven, which appear to cover the range of engines targeted above. Five models of stepper motors and four types of servo (torque) motor actuators were tested. Ultimately, greatest success was obtained with a high-torque servo normally sold for radio controlled boat applications. Learning algorithms were refined to acquire and implement the required smoke limit map for any given engine. Sensor response lag and actuator speed remain known but hopefully minor concerns, dependent upon the time constants associated with a particular engine installation.
The concept of using direct measurement of exhaust smoke for real-time feedback control of fuel quantity is not new, having been previously referred to in a patent\textsuperscript{4} issued in 1986 to United Technologies Diesel Systems, Inc., and in a technical paper from staff at Detroit Diesel in 1991\textsuperscript{5}. The referenced patent was abandoned in September 1998, while under the ownership of Dennemayer and Co. of Heaton Lane, England. To the best of our knowledge, an actual system utilizing any part of this concept has never been commercialized due to two technical obstacles to be described later, both of which have been overcome by the EDSRS.

The most significant technical obstacle was related to the fundamental approach of attempting real-time feedback control of the fuel quantity by direct smoke sensing. In the real-time approach, there is a considerable lag between the detection of the exhaust composition and the actuation of any possible fuel limiting mechanism. This results in control loop instability and periods of smoke during throttle or load transients, even with the feedback control fully operational. The EDSRS does not implement real-time control in the same sense. It is a “learning” system that adaptively updates a smoke-limit map based upon exhaust sensor feedback. The system learns and continuously adapts the smoke limit fuel quantity over time, such that the control actually operates “open-loop” using the adaptive map. The enabling technology behind this learning control method, real-time digital signal processing, was not considered practical only a few years ago.

Figure 1 illustrates the key components of a retrofit EDSRS deployment.

---


\textsuperscript{5} Hartman, Peter G., Plee, Steven L. Bennethum, James E. (Detroit Diesel Corp.) Diesel smoke measurement and control using an in-cylinder optical sensor. SAE International Congress and Exposition, Detroit, Michigan, USA. February 25 - March 01 1991.
As a result of the technical redirection toward a retrofit system, we are not as far along toward commercialization as we had initially hoped, but we have a product that can potentially be applied to a much wider and more needy (from an emissions point of view) market. We note that older mechanically governed diesel engines, lacking any considerations for emission controls, are some of the most egregious polluters of all roadway or off-road vehicles. If the results of our development work can be successfully deployed as a retrofit system for these existing applications, the emission and energy use reduction benefits to society at large would be comparatively much greater than if the technology were (only) used on recent (late-1990’s) electronically controlled diesels, which already are much cleaner and more efficient than previous diesels. Applications will include light duty automotive diesels such as 1980’s vintage Volkswagen Rabbit or Isuzu I-mark or PUP diesel vehicles, the Ford/Navistar 6.9 and 7.3 Liter and GM 6.8 Liter light/medium truck engines, and transit bus and commercial transport engines such as the Detroit-Diesel Allison V53 and V72 series engines. Included in this potential market are almost all diesel-powered passenger cars and light trucks manufactured prior to the introduction of the most recent “clean diesel” engines equipped with particulate traps and electronic controls. Also included are heavy-duty trucks, transit vehicles, school buses, and agricultural equipment. We also anticipate a modest potential market for “diesel engine swaps”, which was popular during the early 1980’s, if the “swapped-in” diesel engine had our retro-fit electronics control which would keep it from smoking. Renewed trends in fuel prices can be expected to drive a resurgence of interest in both new and retrofit diesel engines.

A potentially significant new market for small-displacement diesel engines also exists in hybrid electric vehicles (HEVs). This application typically utilizes an internal combustion engine (ICE), as a primary or auxiliary power source, to charge the vehicle battery (series hybrid) or both charge the battery and power the vehicle (parallel hybrid). The poor emissions characteristics of small diesel engines have generally ruled them out for service as the ICE for a hybrid EV, yet their exceptional energy efficiency strongly recommends them. With adaptive fuel control as mechanized using the method we have developed, the use of small diesel engines as HEV ICES could possibly become attractive.

In continuing to work with this concept, we have learned that the analog waveform of the unconstrained injector needle during an injection event provides additional information of potential value in the feedback control of the injection system. We have long known that the instantaneous injection rate information contained in the waveform was of potential value in reducing emission and idle “knock”, but technology did not permit utilizing it for “shaping” the injection pulse. Recent improvements in injection actuator technology, especially the introduction of electronically actuated individual cylinder “unit” injectors, has opened possible new doors to the utilization of this information. Although our present focus does not utilize these observations, they suggest avenues for future development after initial product introduction.

The use of this method in conjunction with the electronically controlled gaseous fuel injection, either alone or in conjunction with diesel injection, is also an exciting product thrust we have begun pursuing. This applies to natural gas (methane), propane, and hydrogen alternative fuel engines. In brief, solenoid-actuated gaseous injection valves are highly nonlinear devices, which dispense difficult-to-calibrate fuel quantities over their required range of operation. The method we have developed, applied to the electronic gas injector, solves this problem in the same way it does for diesel engines – closing the fuel deliver control loop permits precise fuel deliver over the entire range of operation and life of the engine, independent of parameter changes and component wear. The enabling technology behind our approach, real-time digital signal processing, was not feasible until the introduction of recent high-performance microcontrollers. With the recent resurgence of interest in diesel and alternative fuel powered automobiles and light trucks (driven by increasing gasoline costs), we feel that the market potential and concomitant energy-saving benefits of this technology is more appropriate than ever.

In July 1998, Loragen moved into a new facility in an industrial park at 3576 Empleo St., Unit 1, San Luis Obispo. Two new technical staff were hired: Brian Hemme, MS Electrical Engineering, was employed in December 1997, and assumed a full-time salaried position as senior engineer in July, 1998. Kyle Wayman, a computer engineering student at Cal Poly, was hired full-time for summer 1998, and part-time during the academic year. Plans to incorporate Loragen have been delayed until 1999, due to the accelerated schedule of work on this project.
Progress with our engine-based system development testing was slowed by a change in the availability of the dynamometer-testing facility at Cal Poly that we used for prior work. Our proposed workplan specified the use of this facility for these tests. After waiting over six months for approval from Cal Poly, we requested permission from the DOE to rebudget grant funds to permit us to purchase a dynamometer apparatus to perform these tests. Completion of the dynamometer set up was completed in November 1998. Development and testing proceeded at an accelerated rate since that time.

In its present state of development and commercialization, the system is intended to prevent existing diesel engines from overfueling to the point of visible particulate emissions (smoke), while allowing maximum smoke-limited torque under all operating conditions. An optional exhaust temperature feedback mechanism is available, which adds the valuable feature of engine protection due to excessive exhaust temperature. This latter feature alone may justify the installed cost of the retrofit system, especially for turbocharged engines, since it could potentially save the engine from destruction under excessive load. A component and signal diagram for the retrofit system is shown in Figure 2 below.

---

**Figure 2. Component and Signal Diagram, EDSRS.**
The system employs a microcontroller and exhaust particulate emission sensor to continuously generate and update an adaptive throttle-limit map. This map specifies a maximum allowable throttle position as a function of engine speed and manifold air pressure. The coolant temperature is also used as a parameter, such that full control authority is exercised only after the engine is fully warmed up. The fuel limit is implemented indirectly via the throttle position, since direct access to the fuel rack or sleeve limit is impossible in an aftermarket installation. For mechanically regulated fuel injection systems, the throttle position limit is mechanized via a linear position actuator attached to the fuel injection pump. For electronically regulated injection systems such as our Zexel COVEC electronic pump, this limit is mechanized by interception and modification of the electronic throttle input to the system. Indirect fuel delivery measurement inferred from injector needle lift is optional, to provide a feedback signal for more precise fuel limit-regulation, an especially valuable feature since the actual fuel sleeve or rack position cannot be accessed. In its lower cost embodiment, this system is fully functional but somewhat less accurate without this fuel quantity sensing mechanism. We have developed prototype versions of the system both with and without this feature, each appropriate to different engines and vehicular applications.

One other distinguishing feature of the EDSRS which uniquely qualifies it for retrofit applications is that it functions as an overriding "limited authority" type of control. It operates as a dynamic throttle position limiter rather than as a direct fuel control. This overcomes the problem that in retrofit situations, it is not possible to directly access or limit the fuel delivery means (fuel sleeve or rack) which is internal to the fuel injection pump. Since the EDSRS sets the throttle limit as an instantaneous function of the engine speed (and temperature), it actually parallels the internal function of the mechanical governor, mechanizing the appropriate fuel quantity limit via the only externally-accessible injection control input, the throttle.

As previously mentioned, we are attempting to anticipate the transition to high-pressure direct injection systems in newer diesel vehicles, to retain the applicability of both needle-lift based fuel quantity sensing and direct exhaust particulate sensing to these newer technologies. The present approach is consistent with these developments, as well as developments in diesel particulate trap technologies. As previously pointed out, precise fuel control is an absolute necessity for the survival of any type of diesel particulate trap; a single heavy particulate burst and clog or poison a trap or filter to a degree that prevents on-board regeneration or purging.

With respect to the immediate future and the commercialization of this product, we have made arrangements with Santa Maria Area Transit, a local county transit operator, for the deployment of the EDSRS on one of their in-service buses over an adequate period of time to assess it’s effectiveness, reliability and acceptance by the transit district. We are pursuing similar arrangements with a local school district, for possible demonstration on one or more of their diesel school buses (which presently exhibit a severe problem with smoke emissions under full load). We are in the process of exploring local grant opportunities which would fund the large-scale deployment and testing of this system in one or more of these bus fleets. State of California and local San Luis Obispo and Santa Barbara County funding opportunities have been identified, accessible through the local Air Pollution Control District (APCD), Air Quality Management District (AQMD), and the State-administered Petroleum Violation Escrow Account (PVEA) fund. We intend to actively pursue these options in order to gain additional valuable test data, and, if successful, to create broad awareness of the system and its potential benefits for existing diesel vehicle fleets.
Technical Tasks and Milestones

Development of a practical, reliable real-time exhaust smoke sensor.

The most important element of the feedback system is the real-time smoke sensor. The design of such a sensor is particularly challenging, since it must have a very fast response, and operate at exhaust temperatures reliably and repeatably over the life of the vehicle. We designed, fabricated and experimentally tested several particulate sensing methods for this application. These methods were based upon electrostatic charge transfer, optical transmission and optical side-scatter. We achieved best results with a simple transmissometer, consisting of a high-power infrared LED generator and phototransistor detector, coupled via a circuit that compensates for residue accumulation of the optical elements, temperature effects on the semiconductors, component aging and electronic noise. These elements are isolated from the high temperature and contamination of the exhaust stream by placement in recessed tubes at opposite ends of an 18 inch long “sensor section”, shown in the photographs accompanying this report. Air is injected into the end of each tube to prevent intrusion of exhaust materials into the tubes, and to cool the semiconductor elements. The air supply is provided by a small bleed from a truck’s air-braking system, from a mechanical engine-mounted air pump or a small electrically driven compressor. The sensor section is located in the exhaust plumbing between the exhaust manifold and muffler, as shown in Figure 3 for the naturally aspirated engine. Figure 4 shows a close-up view of the laser light source at one end of the sensor section. For the turbocharged engine, the sensor section is located in the exhaust stream immediately following the turbine outlet.

Figure 3. Sensor Section on Naturally Aspirated Engine.

Figure 4. Close-up of Laser Source at one End of Sensor Section.
Tests with this smoke opacity sensor show that it is sensitive and repeatable, at least during the duration of our dynamometer-based tests. Long-term reliability of the sensor remains a concern.

**Development of a high-torque, high-speed throttle limit actuator.**

The EDSRS functions as an overriding “limited authority” type of control. It acts as a dynamic throttle position limiter rather than as a direct fuel control. This overcomes the problem that in retrofit situations, it is not possible to directly access or limit the fuel delivery means (fuel sleeve or rack) which is internal to the fuel injection pump. A special actuator was engineered to implement the throttle position limit. This application required high torque as well as high-speed actuation, since any actuation lag would tend to destabilize the closed-loop system and decrease accuracy of the control.

We designed and tested several engineering approaches to this actuator problem. Initial designs utilized stepper motors, since these provide exact positioning without the need for a position feedback sensor. After testing seven models of stepper motors from three manufacturers, we concluded that a torque-motor-based feedback positioning design was necessary. A low cost solution to our needs was found in the form of a high-power actuator designed for large radio controlled model airplanes or boats. The Cirrus CS-600 RC servo produces 250 In-oz of torque, with a response time of 0.5 second over 189 degrees of rotation. A cable-intercept actuator was engineered based on this servo, with the addition of limit switches for idle and max throttle limit. Figure 5 below is a photograph of the actuator, installed below the injection pump on the dynamometer test engine. For the commercial prototype, the actuator was integrated into the same case as the controller as shown in Figure 6.

![Figure 5. Throttle Limit Actuator on Test Engine, Located Below Injection Pump.](image1)

![Figure 6. Throttle Limit Actuator Incorporated in Controller Module of Commercial Prototype.](image2)
Computer simulation to aid in adaptive algorithm development.

A computer simulation has been developed as an aid for the development and refinement of adaptive throttle-limit algorithms for diesel engines in general. Preliminary work was done using IBM Quick Basic running on one of our Pentium II/400 PC’s. This simulation duplicates the operation of the diesel engine with respect to its speed, torque and smoke generation as a function of throttle input, vehicle road load and external grade or towing load. A printed copy of the simulation and a disk containing the simulation code (diesim20.bas) are included in the Appendix of this report. The program will run on any IBM-type PC under QuickBasic or most other incarnations of basic. A Pentium II class machine is required or the simulation will run annoyingly slow. For subsequent demonstration purposes, this simulation has been ported for full Windows-95/98 interface using Borland Builder 3.0 C++. This “graphical” version of the program includes an automated driver, and terrain-following over a simulated road course. This version will demonstrate the comparative effectiveness of the EDSRS on virtually any vehicle model. A typical output display from the QuickBasic version of the simulation appears below in Figure 7. This figure shows the simulated ability of the EDSRS to learn and enforce a smoke-limit fuel quantity via a dynamic throttle limit.

![Figure 7. Screen Capture of DIESIM Computer Simulation of EDSRS Controlling a Diesel Powered 9,000 lb. GVW Vehicle.](image)

Final Report, ERIP Project 583
PC-based control development system.

Preliminary to the final design of the microcontroller-based version of the EDSRS, we implemented the complete system using an industrial Pentium-II/400 PC and Quatech DAQ-1201 data acquisition and analog output card. This laboratory system permitted us to display internal variables and exercise degrees of control and modification that would be more difficult on the ultimate target microcontroller system. This also facilitated data collection and analysis directly from engine-mounted sensors. All engine tests to date have used this laboratory system. A screen capture of the PC user display for this control is included with this report. It shows data taken during a series of transient response runs, including engine speed, temperature, manifold pressure, exhaust smoke reported by the EDSRS sensor, exhaust smoke reported by the reference Optimonitor opacity meter, and a graphical depiction of the shape of the throttle limit map “learned” by the adaptive control.

Microprocessor-based production prototype system.

The microcontroller-based target system has been completed, building upon the PC-based development work. After an updated survey of available CPU’s for this application, we selected the Motorola MC68HC12 as the core of the controller, and added peripheral interface components required to mechanize the interfaces. Program code originally written in C++ for the PC is being ported to 68HC12 assembly language for speed and compactness. A copy of the current version of the control code is included in the appendix of this report. Figure 8 shows the EDSRS control module installed on the Isuzu 4FB1 engine on the dynamometer.

Figure 8. EDSRS Control Module Installed on Isuzu Test Engine During Dynamometer Tests.

Dynamometer-based engine tests of complete system.

Engine tests have been completed on the naturally aspirated configuration of the Isuzu 1.8L diesel test engine. The naturally aspirated engine dynamometer setup is shown in Figure 9. Figure 10 shows the engine again, with the reference exhaust opacity measurement equipment mounted to the wall in the background.
Figure 9. Naturally Aspirated Engine on Dynamometer, with PC-based Development System Control.

Figure 10. View of Engine on Dynamometer showing Optimonitor Reference Opacity Meter in Background.

Figure 11 shows a screen capture from our data acquisition and control PC shows typical data for a series of full-throttle acceleration / deceleration experiments.
Figure 11. Screen Capture from PC-based Development System Display Showing Adaptive Engine Map.

The 4FB1 engine was then refitted with a turbocharger to permit testing of the system with elevated manifold pressures. This was deemed important since a large segment of the retrofit diesel market are turbocharged diesel engines. The turbocharged test engine is shown in Figure 12, below.

Figure 12. Turbocharged Isuzu Test Engine on Dynamometer.
Vehicle-based testing.

With the cooperation of Santa Maria Area Transit, an operator of a large fleet of transit buses, we will field test the EDSRS on selected vehicles from their fleet that have known problems with black smoke generation. These are typically the older vehicles in the fleet, which remain serviceable, but which might otherwise be forced to retire due to complaints about exhaust smoke levels. The first vehicle to be tested is a 26 ft. shuttle bus which uses a 5.9 L turbocharged Cummins diesel engine. Field tests are scheduled to begin during summer 1999.

Our previously-acquired 1982 Isuzu I-Mark vehicle, while a perfect receptacle for the Isuzu test engine, has been found to be unacceptable for demonstration purposes due to (1) it’s age and generally poor mechanical condition, and (2) the lack of turbocharging, which does not permit testing the EDSRS on vehicles which have this popular feature. After an assessment of relative market demand by vehicle class, we recently acquired a 1988 Ford F350 with a Navistar 7.3 L turbocharged diesel engine, for product demonstration and test purposes. This vehicle is a typical example of the heavy-duty pickup truck and/or light commercial vehicle market, which represents the largest market segment that could potentially use the EDSRS. Installation of the EDSRS in this vehicle is currently in progress. Figure 13 shows this test vehicle, and Figure 14 shows the Stanadyne Model 80 mechanical distributor-type injection pump used on this engine, in the process of being retrofitted with the EDSRS throttle actuator.

![Figure 13. 1988 Ford Test and Demonstration Vehicle with Navistar 7.3L Diesel Engine.](image)

![Figure 14. Stanadyne Distributor Pump being Retrofitted with EDSRS Throttle Limit Actuator.](image)
Combined smoke control and mid-range particulate reduction studies.

By design, the EDSRS, is intended only to prevent smoke generation due to excess fueling under full engine load. While this is by far the most egregious emission condition for most diesel engines, another important regime for emission reduction is the partial load condition, in which non-visible particulate emissions can occur even with conservatively calibrated fuel limits. Previous research experience has indicated that the induction of a lean homogeneous mixture of some SI fuel, in particular those fuels with relatively wide limits of flammability, is very effective at reducing part-load particulate emissions. Such a solution, often referred to as partial or lean fumigation, could be easily implemented as an additional feature of the existing EDSRS, implemented by the same control and utilizing the same sensors.

Our informal experiments and preliminary literature search suggest the use of methyl alcohol (methanol) for the implementation of an “ultra lean” fumigation scheme, in which a very small amount of methanol is injected under electronics control into the intake air stream. The exact mixture composition would be a function of the present engine operating condition as well as the diesel fuel quantity, since the methanol partially replaces the diesel fuel in the overall charge composition. Prior research has indicated that even for methanol charge compositions as little as $\Phi = 0.10$ (equivalence ratio), significant reductions in NOx are observed, although the effects on particulates are highly variable (possible increase of decrease). If implemented as an optional additional feature of the EDSRS, this would require the addition of a small auxiliary tank of methanol on board, and the installation of a single point electronic fuel injector in the intake air stream. The methanol storage requirement is expected to be approximately 20% of the diesel fuel requirement. This latter factor is considered important for practical consumer and fleet use, since the requirement to “fill up” with methanol every time the diesel fuel tank is filled must be minimized as much as possible.

To continue our studies and preliminary implementation of this additional EDSRS feature, we have received product support from Methanex Corp., and large producer of methanol. We will continue to pursue this research area after completion of the initial version of the commercial EDSRS. We may elect to seek partial funding assistance for this work via the California Energy Commission “Clean Diesel Program” which emphasizes alternative fuel engine replacement or conversion of diesel-powered medium and heavy vehicles.
Actual Project Schedule

Below is a Gantt chart indicating the actual schedule of tasks performed during the grant period.
Details of Hardware and Software Development

Preliminary System Design and Specification of Software Development Tools

Initially, the specific sensors and the actuation system were not determined. The necessary inputs and outputs could not be specified. This pointed to the need for a prototyping environment that would be very flexible and would facilitate rapid development and testing of control algorithms and eventually, the code for the target microcontroller system. There would be a need for debugging tools as well as an easy means to monitor system status to watch the operation of the sensors and actuator in real time.

These requirements led to using a Pentium II based workstation as the development platform, with a high-performance analog interface card for sensor and actuator interface. Borland C++ Builder was selected for the programming environment. The use of Windows 95 as the operating system was done to allow for easy graphical display and development tools such as Builder, at the expense of a true real time OS that would facilitate more timely data acquisition. A major benefit of this development platform is that it allows for graphical display of the system status in real time, a feature that would not be available in a simple microcontroller-based environment. A Quatech DAQ-1202 data acquisition card was used to provide the digital and analog inputs and outputs. This card is able to perform 400,000 samples per second, much more then required to monitor the conditions of a running engine. The DAQ-1202 provides two analog outputs, 32 bits of digital I/O and 16 analog inputs (as well as three timers). Together the PC and data acquisition card provided a base platform to monitor and control the system sensors and actuator.

The engine speed (RPM), engine temperature, exhaust opacity and intake pressure (on turbo charged engines) would be monitored. Engine speed was monitored with a magnetic pickup on the flywheel. Engine temperature was monitored using the existing engine coolant temperature thermistor. Later, we switched to the use of oil temperature as an indicator of the engine temperature, which provided a more accurate indication of the conditions in the cylinder. Intake pressure was sensed with a Motorola MPX2200 integrated absolute pressure transducer connected to the intake manifold of the engine. The actual circuits used will be discussed and shown later. Many methods for determining engine exhaust opacity were designed, implemented and tested. These methods included electrostatic charge transport, laser side scatter, and LED-based beam extinction. These methods will be discussed in detail later.

To tie the actuator and sensor systems to the I/O board on the computer, a set of interface circuits were needed. For the prototype these circuits were bread-boarded and placed in a metal enclosure to provide some shielding from noise. These electronic circuits consist mainly of gain and shifting circuits to provide full scale inputs to the data acquisition card. Figure 15 shows the basic control system and interface block diagram.
Figure 15. EDSRS Control System and Interface Block Diagram.

Throttle Limit Actuator

The actuation system consisted of a means of limiting the amount of fuel available to the engine. As previously discussed, without modifying the fuel pump itself, the only available option is to control the effective fuel limit stop on the pump. One method is to dynamically adjust the fuel limit control screw in the pump, but this method would be limited to a specific injection pump installed on a specific engine. A second and more universal method is to physically limit the maximum excursion of the throttle. This can be done using either a special linkage, or by intercepting the throttle cable before the pump and dynamically limit the cable excursion. We elected this approach.

The actuator is required, in the worst case, to provide enough force to overcome the force of the throttle pedal. To alleviate this force requirement, a spring was added in line with the throttle cable such that the actuator would only have to work against the spring and not the force of the driver’s foot. The strength of the spring only needs to be sufficient to overpower the return spring on the fuel pump (it should not expand under normal operation of the throttle). For the small VE pump on our test engine, this works out to be a force requirement of about 5 lbs. To reduce transport lag problems in the control loop, the actuator needs to be able to transition the full range of movement in approximately one second. With full range equating to about an inch, this gives a specification for a 1 inch/second travel speed at a force of five pounds.

Servo-controlled air cylinders could easily provide the desired speed and strength, but require a high pressure air system and a significant control system to obtain a stable intermediate position. As it turns out, these specifications are right on the edge of available linear stepper motor design. Initial tests using various linear stepper motors from Thompson Airpax and HSI demonstrated that while it was marginally possible, reliability was definitely a concern, especially noting that our test pump is rather small compared those in large trucks.
which we would expect to have stiffer return springs. The cost of linear actuators in this class, either linear steppers or air cylinders, are also on the high side and require a significant amount of external circuitry/hardware to operate.

The cost-effective solution to this problem turns out to be a Jumbo model aircraft servo. The model we used, the CIRRUS CS-600BB, provides 333oz-in (20.8 lb-in) of torque. With a desired control arm of about an inch this works out to over twice the desired strength. These servos can travel a full 180 degrees of rotation in .66sec easily giving more then an inch of travel in a little more then a half a second. Servos are packaged with integrated control electronics. This means all our control system must do is provide a single PWM signal to command position and the servo’s control logic will handle the actual positioning of the control arm. Not only does the servo exceed the necessary strength but it also eliminates the need for additional electronics to control the actuator. Additionally, the cost of these servos as about half one would pay for a linear stepper or air cylinder not including the extra electronics or hardware those devices would require. Due to the simplicity of controlling a servo, only a single digital output line was needed on the DAQ1202 data acquisition card. This port was used to provide a 50Hz pulse that the servo interprets as a position.

The setup of the servo in-line with the throttle cable was previously shown in Figures 5 and 6. This setup works by allowing the throttle cable to rotate freely around the servo on the pulley. When a reduction on the throttle is desired, the servo arm comes in contact with the metal stop attached to the cable pulley. The in-line spring between the driver and the servo allows the servo to reduce the throttle at the pump side without having to force the driver’s foot back. This not only serves to reduce the strain on the actuator but eliminates the odd feeling the driver would experience if the petal were to move in and out with constant foot pressure.

**Speed, Pressure, Temperature, and Throttle Limit Sensors**

The engine speed sensor is a magnetic pickup placed a fraction of an inch from the teeth on the flywheel. The pickup produces voltage spikes for each passing tooth and the frequency of these spikes is used to determine the speed of the engine in RPM. The voltage pulses from the pickup were feed through a frequency to voltage converter such that the speed range of 0 to 6000 RPM became a voltage range of 0 to 5 volts. This output voltage was then feed directly to one of the Quatech data acquisition card’s analog input ports.

Engine temperature was sensed with a semiconductor thermistor inserted in the oil pan of the engine. The thermistor is connected as the bottom leg of a voltage divider, with an adjustable upper leg resistor. The voltage output of this circuit was scaled and level shifted using an op amp, and then feed to the data acquisition card analog input port.

The intake pressure was measured using a Motorola MPX2200 absolute pressure transducer. This sensor is able to measure pressures from 0 to 30psia. A hose barb was placed in the intake manifold on the engine and a small diameter hose was used to connect the transducer to the barb on the intake manifold. The transducer was then interfaced to the data input card with a simple op-amp circuit designed to covert the differential output of the transducer into a voltage with adjustable offset and gain.

Full throttle and idle switches are required as part of the throttle position actuator, to determine respectively when the throttle was against the actuator-enforced limit, and the throttle was returned to the idle position. The full throttle switch allows the controller to know when that the throttle is at the present actuator limit; if the engine is not smoking the control will increase the adaptive map value for this condition to permit additional fuel. The idle switch is used to establish a reference condition at which it is certain that the engine is not smoking. This permits automatic recalibration of the exhaust opacity sensor, which assure sensor accuracy over the life of the sensor, even if the sensor optics become partially coated with carbon from the exhaust.

**Exhaust Smoke Sensor**

The most challenging design obstacle was the exhaust particulate/smoke sensor. This device must be placed in the exhaust stream and report back the level of particulate matter in the exhaust stream. The sensor environment is extreme, not only due to the intense heat but also due to the carbon produced in the exhaust.
Exhaust carbon will easily accumulate on optical sensors making them unusable in a very short period of time. Due to the conductive properties of carbon, it will also short out any sensing mechanisms based upon charge transport in the exhaust path. This requires that the sensor able to withstand extreme temperatures and resist, or be unaffected by, the accumulation of carbon. Several methods were investigated:

**Electrical Charge Transport**

We experimented with the use of electrical charge transport on partially ionized exhaust particles. One initial sensor configuration used a high voltage power supply to establish a high voltage between an anode and cathode in the exhaust stream. Upstream particles are charged, and then discharged at the second electrode or the exhaust tube wall. In the presence of particulate matter of sufficient density, a measurable current will be detected between the two electrodes. This is used as an indication of exhaust smoke. The phenomenon is similar to that of an electrostatic precipitator. After initially attempting to build our own high-voltage power supply with current sense capability, we eventually opted to purchase an off-the-shelf ULTRAVOLT 25A12-P4 high voltage power supply. This module produces voltages from 0 to 25,000 volts with up to 200uA of current from an input supply voltage of 12 volts. Several probes were designed and tested. Typically these consisted of a pointed copper rod inserted in a ceramic cylinder. The cylinder provides heat shielding and electrical insulation. This probe was then inserted into the exhaust stream via a hole in the exhaust tubing. The design of the insertion point also allowed for the attachment of pressurized air used to keep the carbon from building up between the probe and the exhaust tubing wall. Figure 16 shows one version of the upstream electrode being inserted in the sensor section.

This operational phenomena of the sensor was successfully verified, and for short periods of time, the sensor operated predictably and accurately. However, it suffered from carbon accumulation, even with external airflow around the probe. When initially inserted into the exhaust stream the sensor was able to track exhaust particulate mass flow robustly, but after a single high speed run that produced medium levels of soot in the exhaust, the sensor would be partially shorted to ground by carbon deposits on the ceramic insulator. Various lengths and styles of ceramic shielding and metal probes were tested with various amounts of air flow but none were able to achieve the desired reliability that would be required in a permanent vehicular application.

![Figure 16. High Voltage Electrode being Inserted in the Upstream Port of the Sensor Section, for Electrostatic Sensing Method.](image)

**Optical Sensors**

This next category of smoke opacity sensors can be split into two distinct groups: sample type and in-line. The in-line optical sensors attempt to measure opacity by viewing the actual full exhaust stream. Sample type sensors examine only a small fraction of the exhaust taken from the main stream, via a small tap into the main exhaust line.
The in-line sensors have an advantage by looking at the entire exhaust stream. This allows for a very long optical path that allows for relative low sensor sensitivity, with minimum sensor lag. The less sensitive the sensor needs to be, the more reliable the reading. However, there are many drawbacks to looking at the entire exhaust stream. To look at the entire exhaust stream, the sensors must be placed in the main exhaust and therefore suffer from the heat and carbon buildup problems seen in the electrostatic sensor. The task of installing the sensors in an existing exhaust system is not a simple task and requires precision welding of sensor mounts and air inlets as well as removing and replacing a large section of the existing vehicle exhaust system.

A sample type sensor only examines a small portion of the main exhaust. This sample is taken by connecting a hose to the main exhaust line and routing the exhaust to the sensor. The installation procedure is much simpler than the in-line and the heat problems are all but eliminated. However, the fill and drain times of sample-type sensors create a control lag that greatly complicates the control method, and can potentially lead to closed-loop system instability. Also, since sample-type sensors must use a shorter optical path (to reduce chamber size and therefore lag time), they must be much more sensitive than the in-line sensors. This makes them more susceptible to noise. Upon deceleration, the exhaust pressure drops abruptly, possibly below atmospheric briefly, so that the sample chamber is not flushed out unless and external sample pump is used (an expensive addition). This yields incorrect readings of smoke for as much as several seconds after the actual smoke conditions have passed.

Opacity sensing can be done multiple ways. The most straightforward method is to aim a light-emitting source at a light detector and record the beam extinction as exhaust passes between the two. This method will be referred to as light obscuration. Diminished beam intensity at the detector equates to denser smoke (darker exhaust). Another method, called forward scatter, directs a collimated or coherent beam of light at the detector, but slightly off axis. The no smoke condition results in no light at the detector. As smoke passes between the two devices, some light scatters off the particles and falls on the detector. In this fashion the amount of detected light goes up as the opacity of the exhaust increases. A third method, named backward scatter, is accomplished by placing the sensor and emitter in the same location, facing the same direction. Much like the forward scatter case, the sensor will pick up light reflected off the passing particles and scattered back toward the detector and emitter pair. Again this results in an increase in perceived light as the level of smoke increases. The light source for any of the above-mentioned sensors could be either an LED or laser source operating in either the visible or infrared bands. Figure 17 illustrates the three methods that we tested for opacity sensing. Unfortunately initial tests showed that diesel exhaust does not reflect much light in the visible and IR bands. This makes both the forward and backward scatter type sensors effectively unusable. Therefore we selected the beam extinction method for the EDSRS sensor.
Due to the time frame involved, the development system used the most direct and straightforward optical method so that overall system testing proceeded in parallel with sensor refinement. This basic sensor uses an in-line, obscuration type sensor along with large amounts of air flow provided by an external air compressor to avoid carbon buildup on the optics. This initial sensor utilized a 24 inch section of the main exhaust with bends at both ends. At the apex of the bends at the end of the main section a cylinder was welded such that a straight rod could be inserted in one end and out the other. This cylinder serves as a mount for the sensor, transmitter and air inlets. The length of the cylindrical mounts also serve to separate the electronics from the main body of the exhaust tube. This distance, coupled with the forced air, is what keeps the sensor and emitter from being cooked by high exhaust temperatures. The sensor section and laser source were shown previously. Figure 18 below shows the detector end of the in-line sensor, for both the obscuration tests and the forward scatter tests.
Two types of semiconductor laser diodes were used initially. But it was determined that high power infrared LEDs (Light Emitting Diodes) were sufficient and much less costly. The LED light source was powered through a transistor at a constant current. A phototransistor was used on the receiving end to sense the amount of transmitted light. The current through the diode was used to produce a voltage across a resistor which was then scaled and offset before being feed to the data acquisition card. The output of the phototransistor was also used in a closed loop to control the amount of current to the LED. When the engine was in a non-smoking condition (throttle at idle) the current to the LED would be adjusted, either up or down, to produce a specific output voltage at the phototransistor. In this manner, small amounts of carbon buildup could be tolerated without adversely effecting system performance.

While the basic opacity sensing system was sufficient to complete the initial design and take us through testing, a more robust and easier to install version was desired. The ideal sensor would be a sample type because of its ease of installation on a vehicle. While the in-line method would require removing and replacing a section of the vehicles exhaust, the sample type could simply be bolted to the vehicle chassis with a sample line connected to the existing exhaust system. A sample sensor would also solve another problem encountered during testing. When the engine is grossly over-fueling there begins to be luminescence in the exhaust as the unspent fuel is burned traveling down the exhaust tube. The basic sensor will see this illumination as coming from the sensor’s emitter and report a reduction in smoke as the actual level of particulate matter is rising. While this situation only occurs in the most extreme of regimes, it is a situation that must be remedied. The sample sensor would quickly cool the incoming exhaust stream, quenching the luminescent particles before they reached the sensor. Due to the restricted flow the sample type sensors can have problems with latency in the exhaust stream. There are two solutions to this problem. The first is to dump the exhaust into a very low pressure environment, for example having the sample tube exhaust directly to the atmosphere. A second solution is to induct the exhaust through the sample system with a sample pump and then inject the exhaust stream back into the main exhaust line.

Besides allowing the exhaust to cool, there is another means of reducing the influence of luminescent exhaust particles. The transmitted light can be frequency modulated. The receiver then takes the received light energy and filters it for the frequency at which the source was originally modulated. Not only will this reduce the effect of luminescent exhaust particles but this will allow the sensor to work in situations where the devices are not shielded from ambient light sources as long as the phototransistor does not become fully saturated.
Interfacing the optical smoke sensor involves monitoring the output voltage of the phototransistor. As with the other sensors, this output is fed through a simple scale and level shift amplifier, and then tied directly to a data acquisition card analog input.

A block diagram of the complete PC-based development system is shown in Figure 19 below.

![Figure 19. Block Diagram of Development System](image)

**Control and Interface Software**

The control system used in the development stage was written for Windows 95 in C++ using Borland C++ Builder 3.0. This environment allows for visual drag-and-drop type design of the windows used for an application along with the standard compiler features. Builder allows for very rapid development of windows applications and is an invaluable tool for rapid program design.

The PC user interface to the control system was shown earlier. The main window at the top displays the engine’s status in a strip chart format. The most recent data always appears at the right and scrolls towards the left as it ages. The lower window is a three-dimensional plot of the engine fuel limit as a function of RPM and Pressure. This map only applies if the engine is sufficiently warmed up. The controls to the right of the main window allow the user to modify the fuel limit map by hand. This is useful for debugging the map update strategies as well as testing boundary cases that are not easy to achieve with the actual engine. The controls at the lower right allow for direct control of the actuator which is used in system hookup as well as debugging actuator control code. To the left of the main window is the key for the engine statistics plot as well as indicators for the full throttle and idle switches.

System action is controlled with a periodic system timer. Every timer cycle the engine pressure, RPM, temperature, smoke level and switch positions are read and placed in a queue. Two other loops also run, but at slower intervals that are user definable. A control loop runs every selected interval and serves to create and update the fuel limit map. A actuation loop uses the fuel limit map, along with current operating...
conditions, to assign a position to the actuator. The actuator will receive the new position on its next control cycle, which run at a fixed frequency of 50Hz. The frequency of the control and actuation loops was made variable to allow for experimental verification of the optimal control loop timing. It is very important that the actuation loop is not so slow in acting, compared to the control loop, that the map is modified in a major way before the actuator has a chance to react. The timing of the control loop is also critical, if run too fast the map will update far too quickly and the driver will fell oscillations in engine speed as well as see puffs of smoke as the fuel limit map oscillates past the smoke limit. If the control loop is run too slow, the system will never be able to update portions of the map that are only momentarily accessed in transients, such as acceleration from stop to normal cruising speed.

The fuel limit map is only modified by the control loop under two conditions. First, when the engine is producing unacceptable amounts of smoke and the engine is warmed up. Secondly, the map is modified when it is not currently smoking and the engine is at full throttle. The process of updating the fuel limit map is quite involved. The main concern is system lag. The smoke level read by the smoke sensor was not produced by the RPM, pressure and temperature readings that were obtained at the same time. It takes a finite amount of time for the smoke to travel from the cylinder in the engine to the location in the exhaust with the sensor. This smoke lag time also varies directly with the engine speed and pressure. Therefore, to accurately predict what RPM, pressure and temperature conditions produced the current smoke reading one must go back through the RPM and pressure queue and calculate what set of values in the past produced the currently recorded smoke. Once the proper lag time is calculated, the central location in the map to modify is calculated from the saved RPM and pressure readings.

Once the correct map location to modify is located, the map must be altered in some fashion to take into account the new information without ruining older, valid information. As the map is composed of discrete points, many valid combinations of RPM and pressure will result in points in-between actual points in the map. Therefore, to change the fuel limit for an intermediate RPM and pressure point requires changing the surrounding four points. One map modification method would be to only modify the surrounding four points, (or the actual point if the RPM and pressure hit a discrete point). While this method would result in a valid points within the space defined by the four discrete points, the map outside this area would be unaffected by this map modification. The drawback to this method is that the map can develop areas that are correctly shaped next to areas that have not yet been shaped with sharp transitions between the two. This is a very undesirable condition because we know the fuel limit map is a smooth, continuous, monotonic increasing manifold. The desired map adaptation method should task this into account and preserve these map characteristics.

The final, optimized, map adaptation strategy can be envisioned as hitting a metal sheet with a uniquely shaped mallet. This metal manifold lays over the pressure-RPM plane and its height is the fuel limit. When hammering down on the manifold (reducing the maximum allowed fuel quantity), a mallet that is flat in the directions of lower pressure and RPM, but parabolically increasing for increasing RPM and pressure. For the increasing RPM and decreasing pressure or the increasing pressure and decreasing RPM the mallet is only parabolic in relation to the increasing variable. For hammers up on the manifold the mallet in flat in the increasing directions and parabolic downward for the decreasing directions. This is done so that even with hits in only a few selected areas the fuel limit map will start to take shape. While this modifies areas of the map not currently being explored, it does so in a valid way. The fuel limit map is monotonically increasing with respect to increasing RPM or pressure. This means that forcing lower pressure and RPM points to be at or below higher ones when hammering down with the flat portion or forcing higher pressure and RPM points to be at or above lower ones when hammering up is always valid. The parabolic shape on the other side of the mallet was chosen because it best approximated the shape of most engine fuel limit maps and would achieve the correct fuel limit map shape with the fewest “hits”. The “strength” of the mallet “hit” can be adjusted depending of the severity of the smoke observed. In this fashion heavy smoke will cause a quick map reaction, while light smoke will illicit small map modifications to slightly push the map into the correct position.

While the control loop is used to setup the fuel limit map, the actuation loop looks up the current position in the fuel limit map and sets a position for the actuator accordingly. While the fuel limit map is composed of discrete points in the pressure/RPM space, the actuator needs to react to any set of pressure and RPM in a smooth continuous fashion over the entire pressure/RPM space. To achieve this, the fuel limit position that
the actuation loop calculates is based on a linear interpolation between the surrounding four points. In this manor, the discrete fuel limit map can be used to produce the continuous map needed for positioning of the fuel limit actuator. At this point, one can see that the actuator will follow the fuel limit of the engine without regard to the actual throttle position. What this means is that if a driver is initially accelerating slowly and then stomps on the gas, the actuator will already be in position and waiting to retard the throttle as necessary for that given combination of RPM and pressure. If the actuator waited until a smoking condition was present to limit the maximum fuel quantity, there would be puffs of undesirable smoke visible before the actuator could react.

From Development System to Production Prototype

For the most part, the development platform design was rolled right into the final design with few changes. While the PC based development platform was sufficient to develop and refine the control system, it is not practical to install in a vehicle. The goal was to develop a final version that would run on a microcontroller-based system and be no larger then a small shoe box. The target microcontroller chosen was the Motorola 68HC12. This processor is an improved version of the classic 68HC11 with faster clock speeds and additional instructions and integrated peripherals. The reasons for choosing the 68HC12 are its wide industry acceptance resulting in a plethora of development tools, and it’s wide availability at a relatively low cost. While the 68HC12 has many integrated peripherals that reduced the required external hardware, it has a rather limited amount of memory. The 68HC11 has 1K bytes of RAM, 768Bytes of EEPROM and 32K bytes of Flash EEPROM. Because the fuel limit map must be fit into the 68HC12 RAM space the map had to be kept under the 1K boundary including the memory required for other system variables as well as the system stack. With these limits in mind, the size of the map in the development platform was kept within these limits to ensure that the transition to the microcontroller-based system would not result in any unforeseen hardware limitations.

The initial microcontroller-based prototype system included the controller, electronics and actuator in a single enclosure. This 5”x7.5”x3.5” enclosure can easily be fit under the hood of any vehicle. System sensors are connected via the three connectors on the side, with the actuator interfaced by inserting the throttle cable through the top of the enclosure and around the servo pulley. System power is delivered from a single 12V DC line straight from the vehicle’s battery. Intake pressure is monitored by connecting the hose barb on the side of the enclosure to a fitting on the intake manifold with a rubber hose. A RS232 serial port is also externally available through the DB9 connector below the throttle cable insertion points. This communications interface is used to both program the 68HC11 and to provide the communication necessary for debugging the running system.

System Power

The 12 volt power input line is regulated to various internal voltages with a series of regulators and relays. The microcontroller accepts a 5 volt input supplied by a small 5 volt regulator. This regulator is connected directly to the 12 volt input line to assure that the controller will receive power as long as the vehicle battery is connected. A 4.5 volt battery is also connected in parallel with the vehicle power to insure that the system will not completely loose power even in the event of a vehicle battery replacement. The other controller systems are powered through a relay that is controlled by the microcontroller. The relay will only be closed when the system is in full operation so that the system uses negligible power when the vehicle is off. The servo uses a high current 6 volt source supplied by an op-amp regulated power transistor. The system sensors and LEDs run off an internally supplied 6 volt regulator. This is done because the op-amps can only pull up to one volt less then supply voltage. Using a 6 volt source allows them to achieve an output range of 0 to 5 volts, utilizing the entire input range of the 68HC12’s analog to digital ports. The control system senses both the 12 power line, and an auxiliary line connected to the vehicle’s key switch to differentiate when the vehicle is on or off. If the vehicle is off, the 68HC12 will open the relay, turning off the system sensors and then put itself to sleep. In this mode the system uses only little power. In fact, the power used is in the same ballpark as the amount of power the battery leaks if left unconnected. Once the controller senses the vehicle power coming up, it closes the relay and reinitialize the system actuator. This process should take far less time than required for the engine glow plugs to warm up.
Production Sensors

The sensors are basically the same as discussed for the development system. The additional functionality of the 68HC12 allowed for reduced design complexity in some cases. In the following sections a description of each sensor’s operation along with a circuit diagram will be provide for each sensor.

The engine speed (RPM) sensor was one that benefited from simplification due to added 68HC12 functionality. A pulse accumulation port is available on the 68HC12. This port makes it very easy to count the number of voltage pulses and return a frequency based on the period over which the counting took place. To connect the magnetic pickup to the 68HC12, all that was needed was a 5 volt zener diode to provide the necessary overload protection for the input pin. Engine speed can easily be calculated based on the number of pulses accumulated at the port over a fixed interval.

![Figure 20. RPM Interface Circuit.](image1)

Intake manifold pressure is measured with a MPX2200 connected as shown in the following diagram. This circuit is a slightly modified version of a Motorola reference design. Two potentiometers provide gain and scaling to achieve full scale inputs to the microcontroller and allow for easy initial calibration. This circuit is designed to give good linear response over the range of 0 to 30psia (the full specified MPX2200 range).

![Figure 21. Pressure Circuit.](image2)

The temperature sensing circuit remains unchanged from the design used for the development system. The variable resistance sensor (thermocouple) is placed in series with a fixed resistor connected to a 6 volt supply. The resultant voltage between the two resistances is fed into an op-amp. The other op-amp input is a buffered offset voltage from a potentiometer. Adjusting this pot accomplishes setting the baseline temperature while modifying the gain potentiometer allows for setting the full scale output temperature.
Full throttle and idle as well as system disable are sensed by identical circuits. The actual sensor used is simply a contact switch. The following circuit is designed to eliminate switch bounce, protect the microcontroller input from over-voltage and filter and line noise. The resistor provides a stable high to the Schmitt inverter when the switch is open. The diode is used to protect against over-voltage and the capacitor protects against noise and provides half of the debouncing functionality. The Schmitt inverter provides the other half of the debounce as well as insuring that the signal to the 68HC12 is a clean high or low.

One circuit that was not needed in the development system is the voltage sense circuit. This circuit watches the battery voltage and the vehicle key switch to determine if the system should be on or off. This signal is used to determine when to put the system to sleep and is also used to wake up the sleeping controller. The two transistors provide a Boolean AND of the two input lines. When both the 12volt battery line and the key switch line are at 12 volts both transistors are on forcing the line to the first inverter to be low. The Schmitt inverters are used to provide a clean signal to the microcontroller. Two inverters are needed because a low signal is required to cause the microcontroller to awake. This is the only block of circuitry powered from the same supply, with battery backup, as the microcontroller. This is necessary because this circuitry must remain powered when the controller is sleeping in order to wake it back up.
Later in the design process it was determined that it would be necessary to halt most of the system's operation while the vehicle's starter was in operation. Starters tend to use a very large amount of current and can cause huge changes in system voltage. To determine when the vehicle's starter is engaged a "starter sense" line was added. The line is connected to the wire that gives power to the starter solenoid/relay. The resistor and zener diode are used to protect the port which expects input voltages to be no more than 5 volts.

By far the most complex of the electronics, the smoke sensor interface circuit has gone through many different design revisions. While the following circuit is by no means the only option, it is reliable and was used for most of the testing. Work is still being done to find the optimal smoke circuit and sensor configuration for a commercial product. The circuit discussed here uses a constantly illuminated source and monitors attenuation of the received light energy to determine the density of smoke between the sensor and emitter. This circuit employs feedback to determine to proper level of illumination. During periods when it can be safely assumed that no smoke is present, the circuit will examine the received light energy and then increase or decrease the current supplied to the LED to achieve a preset return value.

This circuit can be broken into 4 blocks. These blocks are the transmitter, receiver, integrator and amplifier. The transmitter block consists of a pnp transistor driving the infra-red LED light source. The amount of current supplied to the LED is controlled by the signal coming from the integration unit. The receiver circuitry consists of phototransistor which is connected to the positive power supply through a variable resistance. This resistance is modified to adjust the sensitivity of the receiver. The output of the phototransistor is connected to a voltage follower to eliminate loading of the receiver. The integration unit is an op-amp connected to linearly charge a capacitor based on the difference between the received light level and the preset voltage reference. The relay is controlled by the microcontroller, and is used to let the integrator recalibrate the system during no smoke conditions. When the received light is to intense the integrator's output will rise causing the level of the transmitted light to drop. Conversely, when the received light is not
intense enough the integrator’s output will drop causing the level of the transmitted light to increase. The final block in the smoke sensor circuit is the filter/amplification stage. This circuit takes the output of the receiver and performs a first order low pas filter. This filtered signal is then gained and shifted to adjust the desired smoke observation range into a 0 to 5 volt output. In later versions of the design the relay and integrator block was removed and the LED was driven by a Pulse Width Modulation port of the 68HC12. The pulsed output was fed into the base of a transistor and the output was RC filtered before it was delivered to the base of the LED drive transistor. Both circuits are shown below.

![Smoke Sensing Circuit with Auto-zero Feature](image.png)

Figure 26. Smoke Sensing Circuit with Auto-zero Feature.
Figure 27. Smoke Sensing Circuit with Processor Control

Power Supply and Instrument Interface

Servo power is supplied by the following configuration. This simple circuit takes a 6 volt reference signal and keeps the output of the power NPN transistor at the same voltage level. This circuit is able to supply up to 20 amps of current, more than enough to run the servo even under extreme conditions. The power transistor uses the entire system enclosure as a heat sink and, therefore, runs rather cool, even under heavy load.

Figure 28. Servo Power Circuit.

Included in the final system is a status LED which is meant to be mounted on the dashboard next to the system disable switch. To protect the microcontroller output pin, a 2.2k ohm resistor is placed between the port pin and the line to the LED. In this way the LED can be mounted on the dashboard by itself (no dropping resistor required) and the microcontroller is protected in case the LED signal line is inadvertently shorted to ground, or connected to a 12 volt supply line. This simple circuit is shown below.
Microcontroller Utilization

Of the eight available analog inputs on the 68HC12 only 4 are being used. These are used for sensing smoke, pressure, temperature and the internal smoke feedback signal. The RPM signal is fed into the Pulse Accumulator input pin. Four of the general purpose I/O pins are used for inputting the three switch inputs and the vehicle starting sense line. Two pulse width modulation outputs are used. One for the servo control and a second to control the strength of the smoke circuit's LED. The vehicle power on circuit is connected to the XIRQ pin on the 68HC11. When in sleep mode a shift from high to low on this pin will cause an interrupt forcing the microcontroller to awaken. During normal operation this pin can be read as a normal digital input to watch for when the vehicle is turned off. Two additional pins are used for digital outputs. These outputs supply the status LED drive and the system power relay control.

System RAM is used for program variables, system stack and for storing the fuel limit map. Because the system never loses power once it is installed, the contents of RAM will never be lost and the map will not need to be created each time the driver starts the vehicle. The EEPROM is used to store basic system operating parameters. EEPROM is used for these adjustable parameters so that in the event the system did loose all power it would still retain the basic setup information and be able to recover. These include the full and idle position for the actuator, default pulse width for the smoke sensor LED, as well as the timing variables for the control loops. These values are updated in the EEPROM only under specific conditions. When the system is initially installed, the setup procedure will load values into the EEPROM. Only major changes to the vehicle itself, such as a new fuel pump with a different cable travel range, would cause the controller to make changes to the EEPROM data. During system initialization, when the controller first awakens, system diagnostics determine things such as cable travel range and these values would be modified, if necessary, at that time. Flash EEPROM on the 68HC12 is used to store the static controller code. While the Flash can be reprogrammed in system, it is only specified to have a 100 cycle life expectancy. While a large block of memory, it can not be used to store anything other then static code and data. The only time this area of memory would be rewritten would be during initial system programming or a complete controller code upgrade either to add new features, or to fix a bug.

Control Code Development

All programming for the 68HC12 controller was done in assembler. The complete code listing is included in an appendix at the end of the report. A block diagram of the control code is also provided. The 68HC12 control program operates slightly differently from the way the PC based control system operated. The ability of the microcontroller's separate subsystems to perform their respective operations concurrently allow for a much more efficient execution of the control algorithms.

Two good examples of the microcontrollers ability to perform different task simultaneously are its pulse width modulation (PWM) output and analog input ports. The 68HC12 is able to provide an adjustable frequency and pulse duration PWM signal without and CPU intervention. Once the processor sets the frequency and pulse duration the PWM unit will continue to produce the desired waveform without further CPU intervention. If the processor desires to change and of the PWM operating parameters, all that is required is a quick write to a single control register. The PWM unit will acknowledge the waveform changes at the start of the next
period. The analog to digital system on the 68HC12 works in much the same way. Once the system is setup, it will continue to perform conversions on the selected analog ports at the conversion frequency chosen when the system was initialized. In this fashion the 68HC12 can get updated analog sensor information by just reading a memory where the results were stored. The analog subsystem also provides a register to inform the main processor when new results were written to memory.

When the microcontroller is first powered it will perform basic initialization functions. These functions include setting up the system stack, configuring the serial, analog and timer subsystems, and configuring the processors ports. Once this is complete the fuel limit map is cleared to all maximal values and the engine status queues are cleared.

Once the system has setup it’s internal working environment, it checks the status of the vehicle’s power system. If either the external 12 volt battery line or the vehicle’s key switch are off, the microcontroller will put itself to sleep until the battery is connected and the vehicle is turned on. If the vehicle is turned on and the power is connected correctly, the microcontroller will turn on the other control system electronics. The microcontroller will then load the system default operating parameters from the EEPROM and transfer them to the correct registers in RAM. The system status LED and timer subsystems are then turned on.

To make sure that the cable travel range has not changed since last operation, the system now steps the actuator toward the idle (no throttle allowed) position. While moving the stop closer to idle the idle and full throttle switches are watched. When both the idle and full throttle switches become closed the microcontroller knows the actuator has contacted the fuel cable and pushed it completely to the no throttle position. The position is then saved as the idle, or no fuel, position. The system is now fully initialized. The processor now turns on interrupts, allowing all the microcontroller subsystems to begin operation. The main loop now begins to go through a large loop checking the system power, watching for the vehicle starting, and outputting a user message on the serial line. If this loop sees the vehicle beginning to start or the power to drop off, the processor takes the necessary action to handle the situation. During this main loop the processor will reset a “watch dog” timer. If this module is not periodically reset it will reset the processor and save the state of the machine before it was reset. This allows the system to recover if a glitch caused it to go into an undefined state.

**Operator Interface**

This main loop also handles processing messages received over the serial line. These messages are used to provide debug functionality to the microcontroller. They result in either information dump over the serial line or a change in system parameters. The following command set is supported:

<table>
<thead>
<tr>
<th>Command sequence</th>
<th>Operation performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>Request a dump of the full fuel limit map over the serial line</td>
</tr>
<tr>
<td>D</td>
<td>Disable the control system (sets actuator to full fuel and stops map updates)</td>
</tr>
<tr>
<td>E</td>
<td>Enable the control system</td>
</tr>
<tr>
<td>H</td>
<td>Halts map updates (control system operates normally but with no map changes)</td>
</tr>
<tr>
<td>R</td>
<td>Resume map updates</td>
</tr>
<tr>
<td>A+@</td>
<td>If system disabled, this increases the full throttle position (used for initial setup)</td>
</tr>
<tr>
<td>AA</td>
<td>If system disabled, this decreases the full throttle position (used for initial setup)</td>
</tr>
<tr>
<td>#N</td>
<td>Modify the number of periods per control cycle to #</td>
</tr>
</tbody>
</table>
##P Change the control system period to ##

#X Change the maximum change in RPM that forces a control cycle

#W Modify the cool/warm temperature threshold

#T Change the smoke limit

#I Adjust the smoke LED PWM timer period

L List the EEPROM memory (show the current value of above parameters)

S Save currently modified parameters to EEPROM if changed

Notes:

# represents an 8 bit number (0 to 255)

## represents a 16 bit number (0 to 65535)

The commands can be entered in upper or lower case

When the processor receives a command over the serial line it will respond immediately following the last character that composes the command. No command termination character (such as a return) is required.

The watch dog timer or COP (computer operating properly) system will cause an interrupt if it is not periodically reset. This interrupt service routine will clear a space in EEPROM and then save the current system status to this area. The current system status is composed of the contents of all the processors registers. Once these values have been saved, the COP service routine will then reset the processor and start execution back in the main loop right after the initialization of the fuel limit map. This way all the system variables are reset but the fuel limit map remains unmodified.

One of the microcontroller’s timers is set to interrupt the system at a periodic interval. This interrupt tells the processor to check the idle switch. If the throttle is at idle, and the engine’s RPM is higher than its idle position (meaning the driver let off the gas and the vehicle is decelerating) the processor assumes that the engine can not be smoking and adjust the control signal to the smoke circuit LED to achieve a desired 0 percent opacity at the receiver. This allows the system to continue to operate properly even as component values drift with temperature and time.

The core of the control system’s functionality occurs in a different timer interrupt service routine (ISR). The main system timer is set to cause a system interrupt every 10ms (this value is adjustable). The first task this ISR performs is to read the current engine RPM and pressure and store them in the respective queues. The values of the two switches and the temperature are also read but not queued. If the system is disabled the ISR exits at this point. If not disabled the system will lookup the desired actuator position based on the current RPM and pressure from the fuel limit map. This new actuator position is placed in a register so the actuator PWM unit can change it’s output at the beginning of the next period. Next, a decision is made on whether or not to run the control loop. This is the loop that makes changes to the fuel limit map. This loop is run if the RPM has changes significantly from the RPM last time it was run or if a specific period of time has elapsed. If the vehicle’s engine is not significantly warm the control loop is not entered. Before entering the control loop the system also checks to make sure a control loop is not already running; only one control loop is allowed to run at any one time.

When the control loop is entered it immediately saves all the outside information it needs to operate and then turns the system interrupts back on. This allows the processor to handle other tasks while this longer portion of code is being processed. The processor then checks on the amount of smoke currently being observed. If the amount is greater than allowed or the full throttle switch is closed, the system performs a calculation to determine what set of RPM and pressures stored in the queues produced the smoke of the current reading. Once this is determined the fuel limit map is modified to either restrict the amount of fuel at that location on the map (if there was excess smoke) or increase the fuel limit (if the throttle was at full and the smoke was
below the limit). Once the map modifications are made the control loop performs some cleanup and is then exited.

**Operator Interface and Debugging**

In order to recover some of the debug capabilities from the initial development system a PC based microcontroller interface program was developed. This program, shown below, interfaces to the microcontroller over the RS232 serial port using the command set described earlier. This interface also downloads dumps of the fuel limit map and displays them in a 3d plot for easy visual verification of the systems operation. This interface program also integrates some simple serial communication by allowing the user to enter plain text to send to the controller. The controllers response is displayed in the small text window. There are a few buttons that allow for quick access to some of the command set. These are designed to speed and simplify access to the functions that are necessary to install and calibrate the system. The program was written to run in the Microsoft Windows 95 environment. A screen capture of the PC based user interface is shown below.

![Figure 30. Screen Capture of PC User Interface to Microcontroller.](image)

While the final production prototype system still retains the basic structure of the development model, there are a few differences in the two system wide designs. The block diagram in Figure 31 below illustrates the structure of the prototype system.
Engine Test Data

The system was tested extensively on our dynamometer-mounted Isuzu 4FB1 1.8L 4-cylinder diesel test engine. Several repeated cycles of hardware and software development, testing and refinement were conducted. The engine was tested first in naturally aspirated form, then a Garret T-3 series turbocharger was added, and the engine was re-tested in turbocharged form. Figure 32 shows the torque, power output and exhaust opacity of the baseline engine, with the Diesel-Kiki (Bosch-licensed) Type VE fuel injection pump calibrated to factory specifications. As previously observed in our preliminary work supported by the National-Science Foundation, this engine is actually intended to smoke significantly at full load. Opacity readings as high as 50% were observed under maximum load, the regime of maximum fuel delivery. This somewhat surprising observation is consistent within the emission regulations at its time of manufacture, 1982. The reasoning at that time appears to have been that since diesel passenger cars and light truck emissions were unregulated, and the engine was relatively small and underpowered for the application, overfueling would be allowed as a means for achieving the maximum possible power output. Under emission standards in effect in California since 1987, these particulate emission levels would be illegal.
With the EDSRS engaged and set to target 5% exhaust opacity, the data of Figure 33 were collected. The control system, after a sufficient “self tuning” period, accurately maintains a smoke level at or under 5% opacity over the entire RPM range at full load. A reduction in power output is observed due to loss of the (small) incremental torque provided by overfueling.

**Figure 33. Engine Performance and Smoke Emissions with EDSRS Engaged.**

Figure 34 compares the power output and emission of the naturally aspirated engine with and without the EDSRS engaged. All other conditions were fixed. The reduction in full-throttle power output over the RPM range is observed to be approximately 10% over most of the RPM range, and 5% at the peak power point, 4000 RPM. The smoke reduction is significant, from a peak opacity of 52% at 2000 RPM for the uncontrolled engine to 5% maintained over the entire speed range. The significant reduction in smoke with only a minor reduction in torque (and therefore BHP) is predictable since additional fuel beyond the smoke-limit fuel quantity produces a disproportionately small increase in torque per fuel increment. However, the additional fuel contributed exponential to gross particulate emission, since much of the incremental fuel leaves the cylinder as the products of incomplete combustion, with characteristically high in carbon-to-hydrogen ratios. These products are observed as visible smoke, and are known to be particularly egregious health hazards. By limiting fuel delivery based upon real-time and learned observation of the exhaust opacity, the EDSRS maintains a fuel limit which avoids the regime of exponentially high particulate emissions at the cost of a minor reduction in power output.

**Figure 34. Naturally Aspirated Engine: Comparison of Smoke and Power Output with and without EDSRS Engaged.**
Following tests of the naturally aspirated engine, a Garret AirResearch type T-3 turbocharger, matched to the flow requirements of the engine, was installed. The integrated wastegate of the T-3 turbocharger was set by the supplier at the recommended limit of 7.5 psi. It may be noted that the turbocharger was capable of producing up to 14 psi boost at maximum power with the wastegate actuator overridden. To match the increased air flow, the fuel quantity limit of the VE injection pump was increased by rotating the fuel limit screw 180 degrees, equivalent to an increment we believe to be approximately 25%. This incremental fuel quantity is only estimated because there is only an approximate correlation between degrees of rotation of the fuel limit screw and the actual fuel delivery. Our objective was to maintain the same maximum fuel-to-air ratio, under full boost conditions, as the naturally aspirated engine without boost. In fact, since we observed a generally lower smoke value over the entire RPM range, we actually had been a bit too conservative in increasing the fuel limit. Figure 35 shows the power output, torque and smoke emissions of the turbocharged 4FB1 engine without the EDSRS.

![1.8L Engine, Turbocharged, w/o EDSRS](image)

**Figure 35. Turbocharged Engine, Power Output and Smoke Emissions without EDSRS Engaged.**

With the EDSRS engaged and set to target 5% opacity (following a sufficient self-tuning period) the data of Figure 36 were generated. As with the naturally aspirated engine, smoke is significantly reduced over the entire RPM range, while peak power output and torque are slightly reduced due to the elimination of overfueling.

![1.8L Engine, Turbocharged, w/ EDSRS](image)

**Figure 36. Turbocharged Engine, Power Output and Smoke Emissions with EDSRS Engaged.**
In Figure 37, the power output and exhaust opacity of the turbocharged engine are compared with and without the EDSRS engaged. The EDSRS maintained smoke opacity at the target 5% level, at the expense of a 7% reduction in peak power output.

![Figure 37. Turbocharged Engine: Comparison of Smoke and Power Output with and without EDSRS Engaged.](image)

**Conclusions and Direction for Future Work**

With the financial assistance provided by the DOE grant, the proposed “Indirect Sensing Technique for Diesel Fuel Quantity Control” has been developed from an experimental concept into a near-production retrofit system for existing diesel engines. The resulting prototype is designated the Electronic Diesel Smoke Reduction System (EDSRS). It is based upon a low-cost microcontroller, and utilizes a specially designed real-time exhaust opacity sensor. A number of significant engineering obstacles were overcome in the development and engineering process, including problems of sensor reliability, actuator design, and control lag in both the sensor and actuator loops. The learning-type control adaptively implements a smoke-limit fuel quantity by limiting throttle excursion, based upon feedback of exhaust particulate level, engine speed, manifold pressure, and engine temperature. The system has been tested on both naturally aspirated and turbocharged diesel engines. Among the remaining concerns are the long-term reliability of the system sensors, especially the exhaust opacity sensor. This and other in-service operational concerns will be assessed during the in-service testing phase, using two diesel test vehicles representative of large gross-polluting vehicle populations. Extensions of this technology for NOx reduction, possibly integrating water or methanol co-injection under EDSRS control, are envisioned in future work.

The features and range of applications of the production EDSRS are summarized below:

**Features**

- An add-on system for diesel engines that permits maximum power output at any engine speed without smoke generation.
- Limits smoke due to overfueling only. Cannot correct imbalance between injectors or effects of incorrect injection timing.
- Closed-loop intelligent control system gradually learns the true smoke-limit fuel map of the engine.
• Prevents visible smoke emissions under all conditions with negligible effect on torque.
• Eliminates the need to conservatively set a mechanical fuel limit to assure that smoking never occurs.
• Continuously adapts to changes in fuel quality, engine wear, and operating characteristics.
• No permanent modifications to engine or injection system. Easily overridden or removed.
• Low cost and reliable compared to other emission reduction solutions.

Applications
• Medium and heavy duty trucks and buses.
• Diesel passenger cars and light trucks.
• Agricultural and construction equipment.
• Generators and other stationary diesel engine applications.
• Greatest benefit for older mechanically governed diesel injection systems.
• Easy to install on existing diesel vehicles for most injection system configurations.
• Target retail cost under $1,000.
• Designed to be a low-cost alternative to equipment upgrade or alternative fuel conversion.

Bibliography


Gladyshev, S. P. (Chelyabinsk State Technical Univ.) ; Bunov, V. M. ( Chelyabinsk State Technical Univ.) ; Azarkevich, S. L. (Chelyabinsk State Technical Univ.) ; Makarov, P. S. (Interval Ltd.) Microprocessor...


Heinze, Peter (CONCAWE); Hutcheson, Roger (CONCAWE); Kapus, Paul (AVL List GmbH); Cartellieri, Wolfgang (AVL List GmbH). The interaction between diesel fuel density and electronic engine management systems. SAE Publication SP-1206. Society of Automotive Engineers, Inc., Warrendale, Pennsylvania, USA. 1996 SAE International Fall Fuels and Lubricants Meeting and Exposition Conference Location: San Antonio, Texas, USA October 14-17 1996.


Kunberger, Klaus New MTU 595 engine. A workhorse with racing capabilities or vice versa?. Diesel & Gas Turbine Worldwide v 22 n 10 Dec 1990 p 40-43.


Wolber, William G. ELECTRONIC ENGINE CONTROL FOR HEAVY DIESEL TRUCKS: SENSOR AND STRATEGY SHORTCOMINGS. Cummins Eng Co, Columbus, Indiana, USA, InTech v 29 n 7 Jul 1982 p 47-49.

Appendix

1. Selected presentation slides from OIT Exhibition., December 1998.

2. “DIESIM” simulation software for EDSRS controlling diesel engine: source listing, for MS Quickbasic.
Appendix 1. Selected Slides from Poster for DOE/OIT Exhibition, December 1998.

Technology Description

- An add-on system for diesel engines that permits maximum power output at any engine speed without smoke generation.
- Limits smoke due to overfueling only. Cannot correct imbalance between injectors or effects of incorrect injection timing.
- Closed-loop intelligent control system gradually learns the true smoke-limit fuel map of the engine.
- Prevents visible smoke emissions under all conditions with negligible effect on torque.
- Eliminates the need to conservatively set the mechanical maximum fuel limit to assure that smoking never occurs.
- Continuously adapts to changes in fuel quality, engine wear, and operating characteristics.
- No permanent modifications to engine or injection system. Can be removed completely leaving original system unchanged.
- Low cost and reliable compared to other emission reduction solutions.
Applications

- Medium and heavy duty trucks and buses.
- Diesel passenger cars and light trucks.
- Agricultural and construction equipment.
- Generators and other stationary diesel engine applications.
- Greatest benefit for older mechanically governed diesel injection systems.
- Limited potential to improve performance of newer electronically-controlled injection systems.
- Easy to install on existing diesel vehicles for most injection system configurations.
- Target retail cost under $1,000.
- Designed to be a low-cost alternative to equipment upgrade or alternative fuel conversion.
Status of Development

- Recipient of DOE Energy Related Invention Program Grant. Project in third quarter.
- Experimental prototype system completed. System undergoing dynamometer tests simulating all possible operational conditions.
- Fleet trials by two regional transit bus operations and two commercial truck operations to begin February 1999.
- Production prototype expected June 1999.
- Full production and delivery in kit form targeted for September 1999. Specific kits for most popular automotive and truck engines.
- Research and development firm is interested in an automotive aftermarket distribution partner.
System Components

Electronic Diesel Smoke Reduction System uses three sensors as inputs to a control module, which dynamically modifies the throttle position to prevent smoke formation.
Appendix 2.  DIESIM Simulation Program for EDSRS Development

' Simulation of Diesel Engine
' Diesim V20

' Function declarations
DECLARE FUNCTION hamup! (a1!, a2!, a3!, a4!)
DECLARE FUNCTION hamdown! (a1!, a2!, a3!, a4!)

' Array Declarations
DIM a(4, 8) 'Governor map array
DIM b(8)   'Smoke limit map array
DIM c(64)  'Adaptive throttle map
DIM Time(601)  'Time output data array
DIM rpm(601)  'Engine speed output data array
DIM opacity(601) 'Smoke opacity output data array
DIM throttle(601) 'Throttle position
DIM smokefifo(1000)  'Transport lag fifo array
DIM rpmfifo(1000)    'Lagged engine speed fifo array

' Constants and parameters
delt = .002     ' [seconds] Time increment
Kgears = .54    ' [meters/rev] ground distance traveled per engine revolution
'drag = .07    ' [N-sec^2/m^2] force factor on vehicle as a funct of speed^2
drag = 0        ' [N-sec^2/m^2] for dyno test, no wind load
Ktorq = 2.5     ' [N-m/ul] ind engine torque factor as a funct of fuel
tau = 15       ' [seconds] engine/vehicle inertial time constant
Kfric = 1 / 60  ' 1/Kfric has units of [RPM/N-m] steady state engine "gain"
Ka = 1 / (1 + (delt / tau))   ' intermediate variable for engine dynamics
Kb = 1 / (Kfric * ((tau / delt) + 1))  ' intermediate variable for engine dynamics
smoketarget = 5 ' target smoke quantity, in opacity (0-100%)

' Governor Map: Fuel as a function of throttle and engine speed.
' First array index is throttle (0 thru 3) => throt = 0 thru 1.0
' Second index is speed (0 thru 7) => engspd = 0 thru 7000 RPM

' Throttle Index 0:  Throttle = 0.00 (Idle)
a(0, 0) = 40: a(0, 1) = 5: a(0, 2) = 0: a(0, 3) = 0: a(0, 4) = 0: a(0, 5) = 0:
a(0, 6) = 0: a(0, 7) = 0
' Throttle Index 1:  Throttle = 0.33
a(1, 0) = 50: a(1, 1) = 30: a(1, 2) = 15: a(1, 3) = 5: a(1, 4) = 0: a(1, 5) = 0:
a(1, 6) = 0: a(1, 7) = 0
' Throttle Index 2:  Throttle = 0.66
a(2, 0) = 50: a(2, 1) = 50: a(2, 2) = 50: a(2, 3) = 40: a(2, 4) = 25: a(2, 5) =	a(2, 6) = 0: a(2, 7) = 0
' Throttle Index 3:  Throttle = 1.00 (Max Fuel)
a(3, 0) = 50: a(3, 1) = 50: a(3, 2) = 50: a(3, 3) = 50: a(3, 4) = 50: a(3, 5) =
a(3, 6) = 25: a(3, 7) = 0

' Smoke limit map: max fuel quantity prior to visible smoke
b(0) = 35: b(1) = 38: b(2) = 40: b(3) = 41: b(4) = 40: b(5) = 35: b(6) = 30: b(7) =
25

' Initial values for adaptive throttle limit map:
FOR ii = 0 TO 63
    c(ii) = 1
NEXT ii
' Initialization of variables

  dist = 0         ' [meters] Elapsed road distance
lastdist = -1    ' [meters] Last road distance
speed = 0        ' [meters/sec] Vehicle road speed
throtlim = 1     ' [normalized 0-1] smoke-limited throttle position
comthrotlim = 1  ' [normalized 0-1] commanded throttle limit, prior to stepper
motor slew limit
engspd = 5500    ' [Rev/minute] initial engine speed
pindex = 25      ' print flag, print and wait every 10 iterations
smoke = 0        ' [0 - 100%] generated smoke opacity
smokeout = 0     ' [0 - 100%] output smoke opacity
sindex% = 10     ' [unitless] sample period index
s% = 0           ' smoke transport lag circular queue input index
ss% = 0          ' smoke transport lag circular queue output index
runnumb = 0      ' Simulation run counter
gradeload = 100  ' [Newton-Meters] Load due to road grade, or external
throtin = 1      ' Initial throttle input setting
control$ = "ON"  ' Control activation switch

SCREEN 12        ' VGA screen
CLS
"COLOR 4: PRINT " Diesel Engine Dynamic Response Simulation"
"COLOR 14: PRINT " (c) Loragen ATG 1998"
"COLOR 15: PRINT "
"INPUT "Enter Throttle Setting (0-1, default=0): ", throtin
"PRINT ""

DO ' UNTIL INKEY$ = CHR$(27)  

ONEMORETIME:     ' Check and query again if invalid throttle position entered
IF throtin >= 0 AND throtin <= 1 THEN
ELSE
  PRINT "Invalid Throttle Value."
  INPUT "Enter Throttle Setting (0-1, default=0): ", throtin
  PRINT ""
  GOTO ONEMORETIME:
END IF

runnumb = runnumb + 1  
COLOR 4: PRINT " Diesel Engine Dynamic Response Simulation"
COLOR 14: PRINT " (c) Loragen ATG 1998"
COLOR 15: PRINT "
PRINT "Run": runnumb; " Throttle="; throtin; " Grade="; gradeload; " Control="; control$; " Initial RPM="; INT(engspd); " <esc> to quit"
PRINT ""
PRINT "Elapsed simulation time (seconds): 
|......|......|......|......|......|......|
COLOR 10
PRINT " #";

 t = 0          ' [seconds] Elapsed time
k% = 0         ' [unitless] index for all output data arrays
pindex = 25    ' print index

AGAIN:

' Throttle limit from smoke control
IF throtin > throtlim THEN
  throt = throtlim
ELSE
  throt = throtin
END IF

' throt = throtin  ' disable throttle limit actuator (does not turn off mapping)

' Store output in arrays
pindex = pindex + 1
IF pindex >= 25 THEN  ' print to array once every 25 * .002 = .05 sec
  pindex = 0
  Time(k%) = t
  rpm(k%) = engspd
  opacity(k%) = smokeout
  throttle(k%) = throt
  k% = k% + 1
END IF

END IF

ELSE
  throt = throtin
END IF

END IF

t = t + delt  ' increment simulation time
IF t > FIX(t) + .998 THEN
  COLOR 10: PRINT "#";
END IF

IF INKEY$ = CHR$(27) THEN
  PRINT ""
  PRINT ""
  COLOR 4
  INPUT "><Ctrl-C> to Quit, <Enter> to continue: ", temp  ' instant abort
END IF

' 2-dimensional linear interpolation of governor map to calculate fuel
srpm = (engspd / 1000): j = INT(srpm)
sthrot = throt * 3: i = INT(sthrot)
lfuel = (srpm - j) * a(i, j + 1) + (1 - (srpm - j)) * a(i, j)
hfuel = (srpm - j) * a(i + 1, j + 1) + (1 - (srpm - j)) * a(i + 1, j)
fuel = (sthrot - i) * hfuel + (1 - (sthrot - i)) * lfuel

' Smoke generation calculation
smokelim = (srpm - j) * b(j + 1) + (1 - (srpm - j)) * b(j)
smoke = .1 * ((fuel - smokelim) * 5) + .9 * smoke  ' IIR smoke lag
IF smoke < 0 THEN smoke = 0
IF smoke > 100 THEN smoke = 100
IF fuel > smokelim THEN fuel = smokelim + .25 * (fuel - smokelim)'fuel beyond
smoke limit adds less torque

' Smoke and speedindex transport lag, inversely proportional to engine speed
s = s + 1
IF s >= 1000 THEN s = s - 1000
smokefifo(s) = smoke
rpmfifo(s) = engspd
lag = INT(62500 / (engspd + 1))  ' delay of 0.05 sec at 2500 RPM for delt=.002
IF lag > 999 THEN lag = 999
ss = s - lag
IF ss < 0 THEN ss = ss + 1000  ' queue sized for max lag of 100*delta=0.2 second
smokeout = smokefifo(ss)
engspdl = rpmfifo(ss)
' smokeout = smoke  ' test: turn off smoke transport lag

' Periodic control action
sindex = sindex + 1

Final Report, ERIP Project 583
IF sindex >= 10 THEN        ' 10 * delt = 20 ms sample period
    sindex = 0

IF control$ = "OFF" THEN GOTO CONTIN

' 2-dimensional linear interpolation of governor map to calculate fuel
s64rpm = (engspd1 / 100): j64 = INT(s64rpm)

' From map, find apex of parabola for ball-peen hammer manifold shaping
comthrotlim = (s64rpm - j64) * c(j64 + 1) + (1 - (s64rpm - j64)) * c(j64)
smokerror = smokeout - smoketarget

' Efficiently update adaptive throttle limit map
IF smokerror > 1 THEN    ' dent downwards
    l = j64 + 1: flag = 0
    DO WHILE flag = 0
        parabola = hamdown(l, s64rpm, comthrotlim, smokerror)
        IF c(l) > parabola THEN
            c(l) = parabola
        ELSE
            flag = 1
        END IF
        l = l + 1
        IF l >= 64 THEN flag = 1
    LOOP

    l = j64: flag = 0
    DO WHILE flag = 0
        parabola = hamdown(l, s64rpm, comthrotlim, smokerror)
        IF c(l) > parabola THEN
            c(l) = parabola
        ELSE
            flag = 1
        END IF
        l = l - 1
        IF l < 0 THEN flag = 1
    LOOP

END IF

IF (throt >= throtlim AND smokerror < -1) THEN ' dent upwards
    l = j64 + 1: flag = 0
    DO WHILE flag = 0
        parabola = hamup(l, s64rpm, comthrotlim, smokerror)
        IF c(l) < parabola THEN
            c(l) = parabola
        ELSE
            flag = 1
        END IF
        l = l + 1
        IF l >= 64 THEN flag = 1
    LOOP

    l = j64: flag = 0
    DO WHILE flag = 0
        parabola = hamup(l, s64rpm, comthrotlim, smokerror)
        IF c(l) < parabola THEN
            c(l) = parabola
        ELSE
            flag = 1
        END IF
        l = l - 1
        IF l < 0 THEN flag = 1
    LOOP

END IF
IF l < 0 THEN flag = 1
LOOP
END IF

' Get current map limit value for actuation purposes
's64rpm = (engspd / 100): j64 = INT(s64rpm)
'comthrotlim = (s64rpm - j64) * c(j64 + 1) + (1 - (s64rpm - j64)) * c(j64)
CONTIN:
END IF ' end periodic control loop

' Slew rate limit of stepper motor actuator (1% per delt=> full range in 1 sec)
IF comthrotlim > throtlim THEN
  throtlim = throtlim + .002
ELSEIF comthrotlim < throtlim THEN
  throtlim = throtlim - .002
END IF

throtlim = comthrotlim ' test: turn off stepper motor slew rate limit

' Road load and torque calc
IF engspd > 10 THEN
  load = Kgears * drag * speed ^ 2 + gradeload
ELSE
  load = 0
END IF

torq = Ktorq * fuel - load

' Engine and Vehicle Inertial Dynamics
engspd = Ka * engspd + Kb * torq
IF engspd < 10 THEN
  engspd = 10
GOTO PLOTDATA
END IF

speed = Kgears * engspd / 60  ' .45 meters per second = 1 mile per hour
lastdist = dist
dist = dist + speed * delt  ' road distance in meters

IF t <= 30 THEN
  GOTO AGAIN ' repeat until t = 30 seconds
END IF

' Plot Routine

PLOTDATA:

'Screen data
SCREEN 12 'VGA graphics
CLS

'Draw plot axes
DRAW "BM50,50"
DRAW "C2 D350 R650"
DRAW "BM150,405 C2 NU10"
DRAW "BM250,405 C2 NU10"
DRAW "BM350,405 C2 NU10"
DRAW "BM450,405 C2 NU10"
DRAW "BM550,405 C2 NU10"
DRAW "BM45,400"
DRAW "BU100 C2 NR10"
DRAW "BU100 C2 NR10"
DRAW "BU100 C2 NR10"

'Label plot axes
COLOR 14
PRINT "Diesel Engine Dynamic Simulation (c) Loragen ATG"
COLOR 4:
'COLOR 12: PRINT "Run No.": runnumb
COLOR 15
PRINT "Run No.": runnumb; "Throttle=": throtin; "Grade=":
gradeload; "Control=": control$
COLOR 3: PRINT "Smoke": COLOR 14: PRINT "Throttle"
FOR h% = 1 TO 2: PRINT : NEXT h%: COLOR 4: PRINT "6000": COLOR 3: PRINT "100%"
FOR h% = 1 TO 4: PRINT : NEXT h%: COLOR 4: PRINT "4000": COLOR 3: PRINT "66.6%"
FOR h% = 1 TO 4: PRINT : NEXT h%: COLOR 4: PRINT "2000": COLOR 3: PRINT "33.3%"
FOR h% = 1 TO 4: PRINT : NEXT h%: COLOR 4: PRINT "0": COLOR 3: PRINT "0.0%"
COLOR 15
PRINT "^ 0% Throttle Limit";
COLOR 15: PRINT "[Seconds]"
COLOR 9: PRINT "^ 100% Throttle Limit"
COLOR 15

'Plot data from arrays t(.) and y(.)

'Plot RPM
xindex% = INT(20 * Time(0)) + 50
yindex% = 400 - INT(rpm(0) / 20)
cmdstr$ = "BM" + STR$(xindex%) + "," + STR$(yindex%)
DRAW "X" + VARPTR$(cmdstr$)
FOR h% = 0 TO 599
  xindex% = INT(20 * Time(h%)) + 50
  yindex% = 400 - INT(rpm(h%) / 20)
  cmdstr$ = "M" + STR$(xindex%) + "," + STR$(yindex%)
  DRAW "C4 X" + VARPTR$(cmdstr$)
NEXT h%

'Plot Smoke
xindex% = INT(20 * Time(0)) + 50
yindex% = 400 - INT(3 * opacity(0))
cmdstr$ = "BM" + STR$(xindex%) + "," + STR$(yindex%)
DRAW "X" + VARPTR$(cmdstr$)
FOR h% = 0 TO 599
  xindex% = INT(20 * Time(h%)) + 50
  yindex% = 400 - INT(3 * opacity(h%))
  cmdstr$ = "M" + STR$(xindex%) + "," + STR$(yindex%)
  DRAW "C3 X" + VARPTR$(cmdstr$)
NEXT h%

'Plot Throttle
xindex% = INT(20 * Time(0)) + 50
yindex% = 400 - INT(300 * throttle(0))
cmdstr$ = "BM" + STR$(xindex%) + "," + STR$(yindex%)
DRAW "X" + VARPTR$(cmdstr$)
FOR h% = 0 TO 599
  xindex% = INT(20 * Time(h%)) + 50
  yindex% = 400 - INT(300 * throttle(h%))
  cmdstr$ = "M" + STR$(xindex%) + "," + STR$(yindex%)
  DRAW "C14 X"+ VARPTR$(cmdstr$)
NEXT h%

' Plot Throttle Limit Map Vertically as a Function of RPM

' First plot vertical axis limit at throttle limit = 1.0
xindex% = 450
yindex% = 400
cmdstr$ = "BM" + STR$(xindex%) + "," + STR$(yindex%)
DRAW "X" + VARPTR$(cmdstr$)
FOR h% = 0 TO 63
  yindex% = 400 - INT(5 * h%)
  cmdstr$ = "M" + STR$(xindex%) + "," + STR$(yindex%)
  IF 2 * INT(h% / 2) = INT(h%) THEN ' print a dashed line
    DRAW "C9 X" + VARPTR$(cmdstr$) ' color 9 is violet
  ELSE
    DRAW "C0 X" + VARPTR$(cmdstr$) ' color zero is black
  END IF
NEXT h%

' Plot map data
xindex% = INT(400 * c(0)) + 50
yindex% = 400
cmdstr$ = "BM" + STR$(xindex%) + "," + STR$(yindex%)
DRAW "X" + VARPTR$(cmdstr$)
FOR h% = 0 TO 63
  xindex% = INT(400 * c(h%)) + 50
  yindex% = 400 - INT(5 * h%)
  cmdstr$ = "M" + STR$(xindex%) + "," + STR$(yindex%)
  DRAW "C9 X" + VARPTR$(cmdstr$)
NEXT h%

' Set target point at 5000 RPM, throttle limit = 1.0  ' Max throttle (no limit)
comthrotlim = 1! ' Max throttle
s64rpm = 50  ' 5000 RPM * ( 64 increments / 6400 RPM )

' Draw upward-opening parabola, for denting map downward
flag = 0
FOR l = 0 TO 63
  parabola = hamdown(l, s64rpm, comthrotlim, 0)
  xindex% = INT(400 * parabola) + 50
  yindex% = 400 - INT(5 * l)
  IF xindex% < 640 THEN
    IF flag = 0 THEN
      flag = 1
      cmdstr$ = "BM" + STR$(xindex%) + "," + STR$(yindex%)
      DRAW "X" + VARPTR$(cmdstr$)
    ELSE
      cmdstr$ = "M" + STR$(xindex%) + "," + STR$(yindex%)
      DRAW "C6 X" + VARPTR$(cmdstr$)
    END IF
  END IF
NEXT l
' Draw downward-opening parabola for denting map upward
flag = 0
FOR l = 0 TO 63
    parabola = hamup(1, s64rpm, comthrotlim, 0)
    xindex% = INT(400 * parabola) + 50
    yindex% = 400 - INT(5 * l)
    IF xindex% > 160 THEN
        IF flag = 0 THEN
            flag = 1
            cmdstr$ = "BM" + STR$(xindex%) + "," + STR$(yindex%)
            DRAW "X" + VARPTR$(cmdstr$)
        ELSE
            cmdstr$ = "M" + STR$(xindex%) + "," + STR$(yindex%)
            DRAW "C6 X" + VARPTR$(cmdstr$)
        END IF
    END IF
NEXT l

' Draw 5000 RPM axis line for measurement of parabola deflections
flag = 0
FOR l = 0 TO 190
    xindex% = 3 * l + 50
    yindex% = 150
    IF flag = 0 THEN
        flag = 1
        cmdstr$ = "BM" + STR$(xindex%) + "," + STR$(yindex%)
        DRAW "X" + VARPTR$(cmdstr$)
    ELSE
        cmdstr$ = "M" + STR$(xindex%) + "," + STR$(yindex%)
        IF 3 * INT(l / 3) = l THEN
            DRAW "C6 X" + VARPTR$(cmdstr$)
        ELSE
            DRAW "C0 X" + VARPTR$(cmdstr$)
        END IF
    END IF
NEXT l

PRINT "Press Space to Suspend."

' Continuously Alternating throttle input
IF 2 * INT(runnumb / 2) = runnumb THEN 'if last run was even run number BEEP SLEEP 10 ' display plot for this number of seconds 'throtin = 0 gradeload = 100 ELSE BEEP SLEEP 10 ' display plot for this number of seconds 'throtin = 1 gradeload = 0 END IF

IF INKEY$ = CHR$(32) THEN
    INPUT " Enter Throttle (0-1, default=0) or Ctrl-C to Quit: ", throtin 'enter throttle for next run
END IF
CLS

Final Report, ERIP Project 583
LOOP

FUNCTION hamdown (a1, a2, a3, a4)  ' b2*x^2 + b1*x + b0
  '  x = a1 - (a2 + 2)                ' xoffset = b1/(2*b2)
  '  hamdown = .0025 * x^2 + .01 * x + a3 - .0003 * a4 + .01 ' yoffset =
  b1^2/(4*b2)
  '  x = a1 - (a2 + 40)              ' xoffset = b1/(2*b2)
  '  hamdown = .0025 * x^2 + .2 * x + a3 - .0003 * a4 + 4     ' yoffset =
  b1^2/(4*b2)
  '  IF x > 30 THEN x = 30           ' offset = + b1/(2*b2)
  '  y = .025 * EXP(1.5 * x) + a3 - .001 * a4
  IF x > 0 THEN
    y = .02 * x^2 + a3 - .0001 * a4
  IF y > 2 THEN y = 2
  ELSE
    y = a3 - .0001 * a4
  END IF
  hamdown = y
END FUNCTION

FUNCTION hamup (a1, a2, a3, a4)  ' b2*x^2 + b1*x + b0
  '  x = a1 - (a2 - 1.389)  ' offset = + b1/(2*b2) = 2
  '  hamup = -.0036 * x^2 + .01 * x + a3 + .0006 * a4 - .00694 ' yoffset =
  b1^2/(4*b2)
  '  x = a1 - (a2 - 27.78)  ' offset = + b1/(2*b2) = 2
  '  hamup = -.0036 * x^2 + .2 * x + a3 + .0006 * a4 - 2.778 ' yoffset =
  b1^2/(4*b2)
  '  IF x < -30 THEN x = -30              ' offset = - b1/(2*b2)
  '  y = -.025 * EXP(-1.5 * x) + a3 + .001 * a4
  IF x < 0 THEN
    y = -.02 * x^2 + a3 - .0001 * a4
  ELSE
    y = a3 - .0001 * a4
  END IF
  IF y < 0 THEN y = 0
  hamup = y
END FUNCTION