Distributed Control System for Turbine Engines

P. L. Shaffer
General Electric Aircraft Engines,
1 Neumann Way, MD BBC5,
Cincinnati, OH 45215

A distributed control system (DCS) for a turbine engine has been demonstrated and tested, consisting of prototype electronic interface units (EIUs) connected to data and power busses. In the DCS, a central control computer communicated with smart sensors and smart actuators via a 2.5 megabit/sec digital data bus, using the Fieldbus protocol. Power was distributed to the smart devices as 100 kHz, 100V peak AC, allowing light, simple power converters at each smart device. All smart sensors, smart actuators, and cables were dual redundant. The smart actuators received position demand from the central control computer, exchanged data between channels to provide local redundancy management, closed the position loop locally, and reported actuator position to the central controller. Smart sensors converted sensed signals to digital values in engineering units, and performed local built-in tests. Testing of the DCS was done in a closed-loop simulation with an engine model. Frequency response of the DCS was almost identical with the conventional system.

Introduction

A distributed control system (DCS) for turbine engines is a control system in which smart devices communicate via a digital data bus with the full-authority digital engine control (FADEC), as illustrated in Fig. 1. A smart device could be a smart sensor or a smart actuator, or could combine both sensing and actuation functions. Smart sensors convert analog sensor values to digital form, and provide outputs to the FADEC in engineering units, such as degrees F, Hertz, or PSIA. Smart actuators receive a position command from the FADEC, and perform closed-loop position control of the actuator. Smart sensors and actuators report their status to the FADEC, and can carry out functions such as compensation, built-in test, fault detection, and diagnosis. A smart device (sensor or actuator) consists of a baseline device plus an EIU. The EIU has a data bus and power bus interface to the FADEC. The smart device would be a line replaceable unit (LRU) on an engine, and EIUs would be serviced or repaired at a service center or by the manufacturer of the smart device. An engine area network (EAN) is a communications and power network that connects the FADEC and the smart devices. An EAN would probably have one or two separate cable assemblies per channel. Each EAN cable contains a shielded twisted pair of wires for transmission of digital data among FADEC and smart devices, and a shielded twisted pair of wires to carry power from the FADEC to the smart devices.

Design studies have indicated that the benefits of a DCS should include reduced weight for the control system, particularly in the cables; reduced development cycle time and design cost, due to common modules and standard interfaces; accommodating growth in sensors and actuators, making the FADEC design adaptable to new engine types, adaptability to changes in components, allowing the opportunity to introduce new technologies; improved reliability and reduced costs due to longer production runs based on standard components and designs; reduced maintenance costs, due to built-in test and diagnostics at each smart device; and enabling of off-engine FADEC for further weight saving, improved reliability and control system integration.

The distributed control system program defined a DCS architecture that should provide the benefits listed above. The pro-

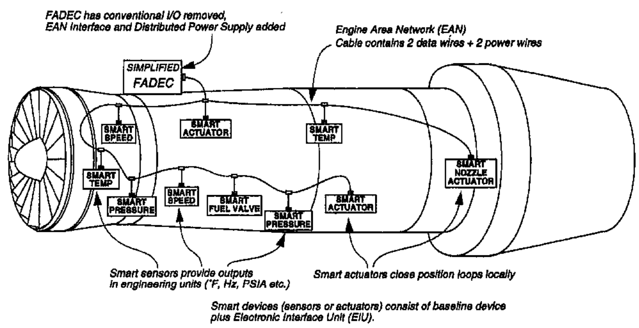

Fig. 1 Overview of a distributed control system on a turbine engine

gram quantified the costs and benefits of the DCS, designed and fabricated elements of a DCS using low temperature parts, and tested and demonstrated a DCS in closed-loop operation with an engine simulation in a laboratory environment. Numerous design decisions relative to a DCS were examined, such as what functions should be carried out by smart devices, whether dual-redundant smart devices should have a local data link between the channels, and what the topology of the EAN should be. This paper will focus on only the following two issues: the characteristics of the power distribution bus and the protocol of the digital data bus. Alternatives will be examined, the rationale described for the, choices made in this program, the results presented, and recommendations made for future distributed control systems for turbine engines.

DCS Design Issues

EAN Power Distribution. Three alternatives for power distribution from the FADEC or central power converter to the smart devices are (1) an AC bus with local AC/DC conversion, (2) a DC bus with local DC/DC conversion, and (3) a central DC/DC power converter generating all voltages, which are distributed to all smart devices. The third alternative can be eliminated immediately because this system would have heavier cables than the other alternatives, would limit EIU designs to use only the voltages produced centrally (or else add local DC/DC converters and thus have the worst aspects of both options 2 and 3), and would make it difficult to ensure good power quality without local regulation.

A DC bus would probably be either 28 VDC or 270 VDC, and an AC bus could be 115 VAC at 400 Hz. All three of these

Contributed by the International Gas Turbine Institute and presented at the International Gas Turbine and Aeroengine Congress and Exhibition, Stockholm, Sweden, June 2–5, 1998. Manuscript received by the ASME Headquarters April 1, 1998. Paper No. 98-GT-16. Associate Technical Editor: R. Kielb.

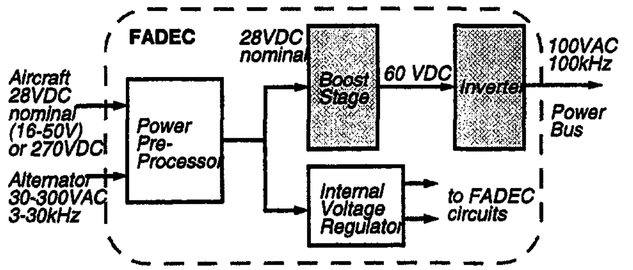

Fig. 2 Central power converter. Shaded portions are additions to existing FADEC power supply.

voltages are described by MIL-STD-704D and are used for power distribution in military aircraft. 270 VDC appears to be the power distribution choice for next-generation military aircraft. However, none of these voltages is very well regulated, or particularly well suited to the needs of a DCS. Instead, AC power distribution using high frequency (100 kHz) and fairly high voltage (100 Volt peak) was selected for this program because it has the following advantages:

— Smaller parts count than DC. Each EIU needs only a multiple-winding transformer with a secondary, rectifier diode, or bridge, and a series regulator for each required output voltage.
— Better stability than DC, because local supplies with series regulators provide a positive impedance to an AC bus while local DC/DC converters provide a negative impedance to a DC bus.
— Better reliability and lower weight than DC, because of the smaller parts count.
— Better high temperature capability than DC, because only simple semiconductors are needed at each EIU.
— High frequency results in smaller transformers, limited by losses in the bus cables and transformer inductance leakage, and by the potential for EMI emission.
— High voltage minimizes current losses in the cables, but is limited by concerns for insulation stress, corona, and personnel safety.

One negative feature of AC power distribution is that the overall power conversion efficiency is somewhat lower than in a DC distribution system, because the efficiency of series regulators in the AC/DC local converters is lower than the efficiency possible in DC/DC converters. The overall efficiency of the AC power system is expected to be about 68 percent, while DC/DC converters can achieve up to 86 percent efficiency (based on a survey of off the shelf, military-qualified DC/DC converters).

A trapezoidal (near-square) waveform was selected (versus a sine wave) because it reduces the peak current flow and low order harmonics (and thus EMI), and is easier to generate than a sine wave. This AC power would be generated within the FADEC or a central power conversion module, and distributed via shielded power distribution wires. A block diagram for the central power converter, which provides well-regulated power to the EAN bus, is given in Fig. 2.

DCS Bus Power Supplies. For evaluation of the prototype DCS, two 100 kHz 100 VAC power supplies were designed and built. Each power supply generated power for one channel of the prototype DCS. Each power supply utilized 60 Hz, 115 VAC power as its input, and generated 100 (±5) kHz, 100 (±5) Volts peak with a trapezoidal waveform. Each supply had a steady-state current output capability of over 2.0 A RMS, with a transient capability of 5.0 A RMS. Each of the supplies was constructed using two commercial off-the-shelf 115 VAC to 100 VDC power supplies as a front end. The outputs of the two DC supplies was switched using current-sensing MOSFET switches to produce the trapezoidal AC output. The bus power supplies were designed to be fully tolerant of any fault condition likely to be imposed on them, such as shorting of the outputs. The outputs were capacitor coupled, and the DC level of the outputs was monitored so that the outputs could be disabled if a fault occurred that resulted in a significant shift in DC level of the output.

EAN Data Bus Requirements. A protocol is a well-defined set of rules and conventions used by the nodes on a network to communicate with each other. A large number of different communication protocols are in use. These requirements for communication via the EAN data bus were defined as follows:

1 Reliability of message content. This requires at least an error-detection code, and preferably an error detection and correction (EDC) code within each message.
2 Assurance of delivery of data. For the FADEC to know that messages were delivered properly to the appropriate EIU, the EIU should always acknowledge the message and indicate whether the message was delivered without any apparent errors.
3 Adequate address range. The FADEC must be able to communicate with an adequate number of smart devices; 256 smart devices should be sufficient for the foreseeable future.
4 Minimum 16-bit data field. Data from sensors or to actuators on engines typically have 12 or 16 bits of resolution, so a 16-bit data field should accommodate system needs.
5 Status and data messages (or status field with data). The smart devices should regularly inform the FADEC of their health status; preferably, at least one status bit should be included with each message; detailed status could be transmitted in separate status messages.
6 Low overhead. Overhead includes bits in a message used for addressing, synchronization and error checking, as well as retransmissions due to collisions or other reasons. Messages in an EAN are expected to be short (one or two 16-bit words of data per message). Some protocols, designed to handle transmission of large messages, impose an excessive overhead on short messages, adding to data latency.
7 Minimum complexity. Because each EIU requires an interface circuit for the EAN, the complexity of the protocol should be kept as low as possible to reduce circuit complexity, cost, and component failure rate.
8 Command-response protocol. A protocol in which the smart devices only communicate in response to a command from the FADEC is preferred, because it is deterministic and less likely to allow failures in which one EIU disrupts the bus via uncontrolled transmissions.
9 Data rate. The minimum required data rate will depend on the amount of overhead in the protocol (and thus the effective data rate relative to the raw data rate), the maximum number of nodes (devices) permitted on the network, and the iteration time of the FADEC.
10 Synchronization. If transmission of data is synchronous with its production and use, then only the transport delay needs to be added to the delay from time of production to time of use. However, if transmission is asynchronous with either production or use of data, then additional delays are added based on the iteration times of production, transmission, and reception of the data.

In addition to these protocol requirements, the electronic devices used to implement the EIUs on engine sensors and actuators must be able to withstand the harsh conditions on engine cases. Most challenging of these conditions are the high temperatures found on engine cases. While temperatures vary among

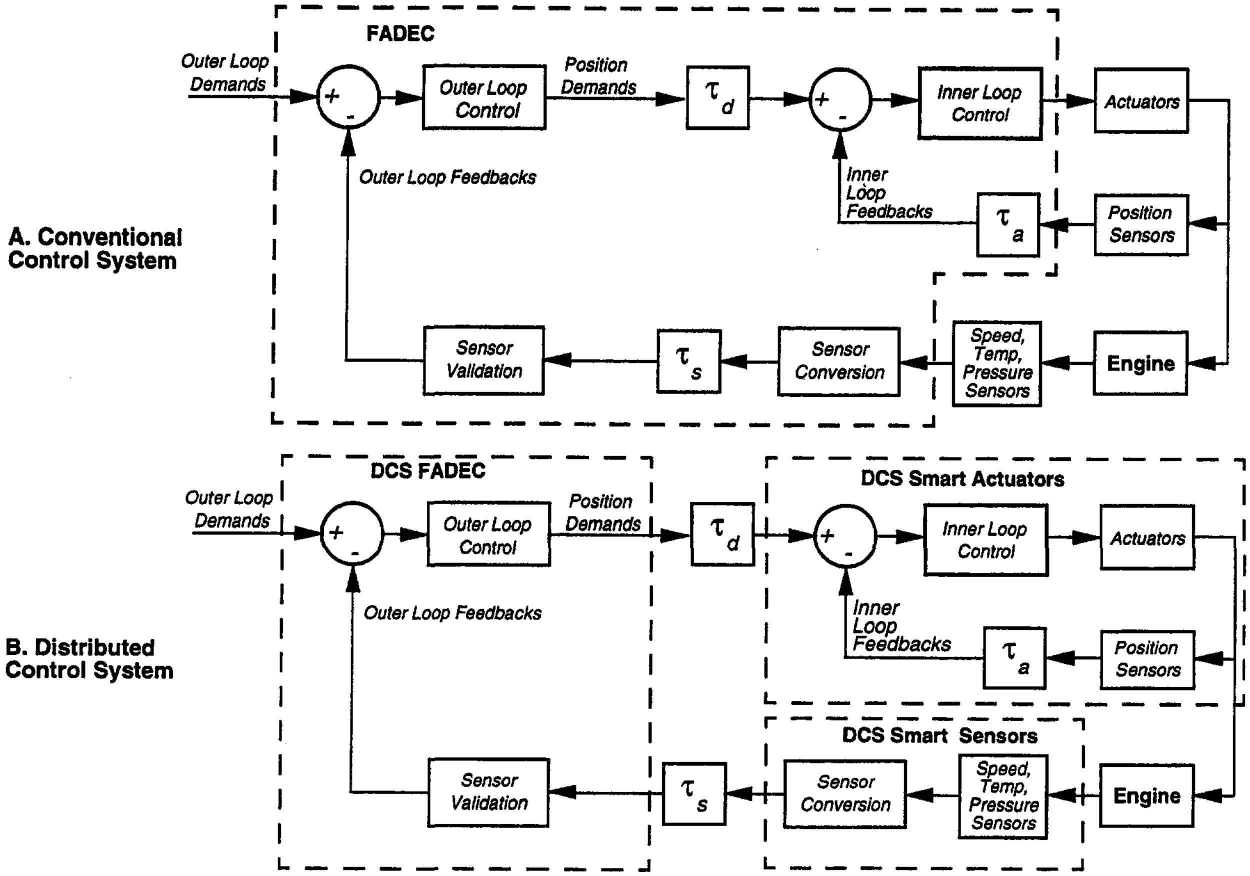

Fig. 3 Delays presented by (*A*) conventional and (*B*) distributed control systems. Areas in dashed boxes represent independent processing units.

engines and among locations on an engine, EIUs will have to operate at temperatures of at least 200°C to be usable over most regions of turbine engines.

Control System Delays. The net effect of data rate, overhead and synchronization can be summarized by the transaction time of the data bus, defined as the time for an interaction between the FADEC and a smart device, in which the FADEC sends a command or request to a smart device, and the smart device responds with an acknowledgment of the command or data fulfilling the request. The transaction time represents a communication delay between the FADEC and the smart device, and added time delays can have a negative impact on control system performance, particularly in reducing system stability.

Delays within a conventional control system and a DCS are illustrated in Fig. 3. Three delays are indicated in this figure: τ_s is the sensor acquisition delay, which includes all input signal processing, such as sensor conversion, validation and selection time; τ_d is the delay from production of a position demand until its use for inner loop position control; and τ_a is the iteration rate of an actuator inner loop control. It is assumed here that the fundamental iteration rate of the FADEC, typically 10 to 20 msec, is the same for the two systems.

Data bus communication delays in a DCS add to τ_s, where sensed engine values are passed to the outer loop control, and τ_d, where position demand values are passed from outer to inner loop control. However, in a DCS, sensor values can be acquired and processed in parallel by smart sensors and a reduced amount of input signal processing occurs sequentially in the FADEC, and inner loop closure computations and actuator output processing can be performed in parallel by the smart actuators. In a conventional control system, computations required for sensor input processing, actuator output processing, and inner loop closure must be done sequentially within the FADEC. Thus, τ_s will include delays due to the sequential processing of multiple sensor signals, and τ_d will include delays due to the sequential processing of multiple actuator signals. The actuator control loop is expected to run several times faster in smart actuators in a DCS than in a conventional control system, so τ_a would be smaller in a DCS (5 msec in this program) than in a conventional control system (typically, 15 msec). The actual impact of a DCS on control system delays will depend heavily on implementation details, but the net effect should be small. In an example control system analyzed, a DCS would have no adverse impact on control system stability relative to a conventional control system for a transaction time of up to 188 μsec.

Data Network Implementation. Existing computer communication protocols were surveyed. Some characteristics of a number of alternative data bus protocols are summarized in Table 1. Included in this table is a custom protocol optimized for engine control applications. This was a minimum overhead, command-response protocol with an EDC code for single-bit error correction. The protocol was designed to operate at 10 Mbps, and was implemented in a breadboard-quality prototype at 2.5 Mbps (due to a limitation in then-available Manchester encoder circuits).

Any of these protocols, except ARINC 429, would satisfy the typical 188 μsec requirement cited above. ARINC 629 can be eliminated from consideration because it is not a command-response protocol. MIL-STD-1773 can be eliminated because it uses an optical interconnect, and prospects for electro-optical interfaces operating at 200°C are poor for the foreseeable future. The protocols in Table 1 that are the most promising are the MIL-STD-1553 protocol, the GE custom protocol, and the

Table 1 Alternative candidate protocols for the EAN

Protocol	Speed (Mbps)	Number of ICs	Temp. Range	Time/ Transaction (μsec)[a]	Number of transmitters per bus
ARINC 429	0.1	2	125°C	674[b]	1
MIL-STD-1553	1.0	1 Hybrid + 1	125°C	72[c]	31
GE Custom	2.5	3 large + 16 small	125°C	28	256
ARINC 629	2.0	1 Hybrid + 3	125°C	83[d]	120
MIL-STD-1773	1.0	Not available	125°C	72[c]	31
Fieldbus	2.5	7	85°C	95[e]	65536

a. Transaction = 16-bit command to EIU from FADEC, plus 16-bit status / value response from EIU.

b. Plus delays introduced by hardware at each end.

c. Response time = 12 μsec (allowed range is 4 to 12 μsec).

d. Increases with number of EIUs; value given is based on 36 EIUs.

e. Delays due to lack of synchronization not included; smart device status is a separate transaction.

Fieldbus protocol. MIL-STD-1553 has the lowest data rate of the three, and has the lowest number of smart devices possible per data bus. But MIL-STD-1553 is widely used in military avionics systems, has low overhead in transmission compared to the Fieldbus, and may have adequate addressing for a turbine engine control system. The GE custom protocol is optimized for engine control, but exists only in a breadboard form, and has little prospect of being developed into a standard chip set. The Fieldbus protocol is an international standard (actually, competing standards), and may have support from manufacturers of smart sensors and actuators. There is some interest from semiconductor manufacturers in development of high temperature (200°C or more) versions of MIL-STD-1553 or Fieldbus chip sets.

The Fieldbus protocol was selected for use in this program and the FullFIP2 microcircuit (Cegelec, 1994) was used to implement the protocol. Table 1 reflects some knowledge gained since the selection of Fieldbus was made, and Fieldbus appeared more advantageous based on information available when the selection was made. Assessment of the success of this choice, and observations on the problems associated with the Fieldbus, are presented below.

The physical layer of the WorldFIP standard is described by ISA-S50.02 (ISA, 1992). Data are transmitted at 2.5 Mbps over shielded, twisted wire pairs. Peak-to-peak voltage of the signal is 5.5 to 9.0 Volts at a transmitter, and must be at least 0.7 Volt at a receiver. A transmission frame consists of a frame start sequence, data and check fields, and a frame end sequence. The frame start sequence consists of an 8-bit preamble, used for receiver synchronization, and a frame start delimiter that occupies 6-bit periods, signalling that a frame is starting. The frame end sequence consists of a frame end delimiter that occupies 7-bit periods, and signals the end of a frame.

The data-link layer of the WorldFIP protocol is described by C46-603 (UTE, 1990). At this layer of the communication protocol, the data and check fields of each frame are defined to consist of a Control byte, a data field of up to 128 bytes, and a 2-byte frame check sequence. The control byte identifies the type of frame. The frame check sequence consists of a cyclic redundancy check code, and is used to provide a data integrity check on the frame contents. The data-link layer provides two types of transmission service: cyclic exchanges of identified variables, and message transfer upon explicit request. Only cyclic exchanges of identified variables were used in the prototype DCS. With variables that are exchanged cyclically, the names and periods of all variables that are to be exchanged are defined at the time the system is configured. Each WorldFIP system has a fixed number of variables that are to be exchanged, and each variable has a unique 2-byte identifier. Each variable can consist of a single value, such as an integer or floating-point value, or can be a composite structure containing a number of values, with a size up to 128 bytes.

A WorldFIP network is composed of two functional types of nodes attached to the transmission medium: a bus arbitrator (BA), and producer/consumer functions. Any node on the bus can carry out producer/consumer functions, but only one node can carry out the BA function. Although the standard allows the BA function to move from one node to another, in an EAN the BA function should always be carried out by the FADEC. The BA controls access to the transmission medium, and is responsible for initiating each periodic variable exchange by broadcasting the identifier of the variable onto the bus in an identifier frame. The identifier frame contains a total of 61 bits, and so requires 24.4 μsec at 2.5 Mbps. One and only one node on the bus is a producer of each variable, and that node responds to the broadcast identifier of the variable by broadcasting the value of the variable in a data reply frame. Any node on the bus that uses ("consumes") that variable then receives the value, having been alerted that the value was imminent by the preceding identifier frame containing the variable's identifier. The BA maintains a list of variables that are to be exchanged cyclically, and scans that list over a fixed period. This ensures that all the variables are exchanged in a deterministic manner, and that real-time deadlines are always met.

The FADEC carries out both BA and producer/consumer functions; however, BA operation is asynchronous with the producer/consumer operations. Since the transmission of the position demand on the data bus is asynchronous with production of the value by the FADEC, and asynchronous with the reading of the demand value by the smart actuator, then the average delay associated with this transmission, illustrated as τ_d in Fig. 3, would be the transport time plus hardware delays, plus one half the periodic transmission interval, plus one half the smart actuator minor frame interval. With a 20 msec transmission interval and a 5 msec smart actuator iteration interval, the average τ_d would be about 12.5 msec. The worst case delay would be almost twice the average value.

Smart sensors and actuators only carry out producer/consumer functions. A smart device examines each identifier frame on the bus. If an identifier frame contains the identifier of a variable that the smart device produces, then the device broadcasts that variable on the bus. On a 2.5 Mbps WorldFIP network, the device should (according to the WorldFIP standard) respond with the data reply frame containing the value within 4 to 28 μsec; this is the turnaround time of the producer node. The turnaround time for the FullFIP2 device operating at 2.5 Mbps is 33 μsec (Cegelec, 1994), which exceeds the maximum allowed by the standard. If a smart device sees an identifier frame containing the identifier of a variable that it consumes, then it receives the following data reply frame, which contains the needed value. This interaction of a smart device with the bus is carried out by the Fieldbus communications processor and does not require intervention of the main processor in a smart device.

The response frame by the producer device includes a total of 61 bits of overhead in addition to the actual data sent by the producer. Efficiency of this protocol increases with the amount of data included in each data reply frame, and decreases with the turnaround time of the producer nodes. With 4-byte variables and a 33 μsec turnaround time by producer nodes, the efficiency of data transfer is 10.7 percent, and 171 cyclic exchanges can take place in a 20 msec period.

The operation of the bus is determined by a configuration file that is processed by a configuration compiler program. This file specifies information about the Fieldbus such as data rate, and information about each of the nodes on the bus. Node information includes type of Fieldbus interface, turnaround time, whether it is the bus arbitrator, and what variables the node produces and consumes. Variable information includes an identifier, periodicity of the variable (in msec), and the data type of the variable. Optional characteristics that can be specified for variables exchanged on the bus are promptness and refreshment; these indicators of data freshness are supplied by the application layer of the Fieldbus protocol. Refreshment indicates whether the application process on the node which produces the variable actually has done so with the periodicity indicated in the configuration file. If for some reason the variable is not generated as frequently as required, the refreshment status will be false. Promptness indicates whether the messages containing a specified variable are received by a consumer with the periodicity indicated in the configuration file. If the bus arbitrator is not requesting the variable often enough, or the producer is not transmitting the message containing the variable with the indicated frequency, then the promptness status will be false. If either refreshment or promptness is false, then valid data have not been received at the desired frequency. In this program, promptness and refreshment were used for the critical variables.

The configuration compiler determines a schedule by which variables are to be transmitted on the data bus, and determines how much of the potential data rate of the data bus is used by the periodic variables. The schedule is compiled for use by the bus arbitrator, to control its sequence of operations. For the configuration file used in the prototype DCS, and representing a fully distributed control system, an average of 4.79 msec of a 20.0 msec frame was used in the transmission of data, or 23.6 percent of the frame. The maximum bus utilization was 24.5 percent of a frame. This indicates that, even with the substantial overhead of the Fieldbus protocol and the 33 μsec turnaround time of a FullFIP2 device, less than one quarter of the available bus data rate was used for data transmission.

Results

Power Bus Measurements. The power supply developed for this program was designed to provide nominal input voltage characteristics to the EIUs. Because these power settings were fixed, testing of the power bus was limited to characterization of its voltage signal. Measurements were taken to record power system characteristics during DCS operation.

Power bus voltage waveform characteristics were measured and compared to the specifications for the power supply. All specifications were met, with rise and fall times being 1 μsec, compared to the specified 3 μsec minimum rise time. No DC component, frequency modulation or amplitude modulation were observed in the power supply outputs.

Additionally, power bus performance was examined for interactions induced by the communication bus. No interaction was visible as distortion or modulation of the power bus waveform.

Communication Bus Measurements. Communication bus performance was examined for interactions induced by the 100 kHz power bus. In the absence of a power signal (i.e., with the power supply turned off), the data signal was a 3 V peak amplitude signal, with the data appearing as pulse-width modulations alternating with intervals of no signal. Minimum apparent pulse width was 200 nsec, corresponding to one half the period of a 2.5 Mbps Manchester-encoded signal. Durations of the pulse modulations, and the intervals between the modulations, varied with the usage of the data bus.

A noise signal at a frequency of 100 kHz was observed on the data bus when the power signal was applied to the power bus. The noise signal was a 100 kHz pulse train, where each pulse had an approximately triangular shape, with a peak voltage of about 1 V, and a pulse width of about 1 μsec. These noise pulses were best observed in the intervals between data transmissions on the data bus. When the noise pulses coincided with data transmissions, the distortion of the data waveform was evident. The pulse width corresponds to the observed rise time of the power signal. These pulses were clearly induced noise from the high frequency components of the power signal, despite the fact that both power bus and data bus were shielded cables. The only place where the two signals were unshielded were the backshells and contacts of the cable connectors, and inside the EIUs.

Smart Actuator Frequency Response. The frequency response of two smart actuator EIUs was evaluated by sinusoidal perturbation of the actuator position demand. This was done at a steady-state engine operating point with FADEC control laws disabled to prevent these control laws from counteracting the perturbations of position demand. The tests were performed on two EIUs with different position control algorithms: a proportional controller and a proportional plus integral (PI) controller. Frequency responses of the actuators were analyzed to produce Bode plots of gain and phase shift of the actuators versus frequency of the input perturbation. The bandwidth of the proportional control actuator decreased from 23 rad/sec for a reference model to 18 rad/sec for the DCS actuator. For the PI control actuator, which was underdamped, bandwidth increased from 25 to 30 rad/sec. For both actuators, the phase shift of the DCS actuator was larger than the reference model. At 10 rad/sec, the phase shift of the proportional control EIU increased from -5 to -7 degrees, while for the PI control, phase shift increased from -37 to -58 degrees. These changes in frequency response are ascribed to the communication delay, τ_d, which is primarily due to the lack of synchronization between production of the position demand and actual transmission under BA control.

System Frequency Response. The frequency response of the DCS system was evaluated by applying sinusoidal variations in power-lever angle command, and examining thrust of the engine simulation as a function of frequency of the input perturbation. Bandwidth of the DCS was about 9.6 rad/sec, versus 9.7 rad/sec for the conventional control system, and phase shift was unchanged between the conventional control system and the DCS. Thus, even with measurable changes in smart actuator frequency response, the response of the overall DCS did not change significantly.

Conclusions

Power Supply Issues. Using the 100 kHz power supplies, 100 kHz noise was visible on the data bus, although interference was not apparently a problem. The primary approach to alleviating this noise should be to improve the shielding of the EAN power bus from the EAN digital communication data bus. Efforts in this area should center around reducing the susceptibility of the communication bus to radiated noise generated by the power input, such as improved shielding in connector backshells, in the connector pins, and inside the EIUs. Lowering the frequency of the AC power source and changing the waveform of the AC power from a near-square wave to a sine wave would reduce the radiated emissions. However, use of a lower frequency would require larger transformers for comparable power supply efficiency, and both changes would require larger filter capacitors for comparable output voltage ripple. Thus, a change in frequency or waveform should be made only if power bus interference with the data bus cannot be resolved with shielding.

Power supplies in the EIUs for this program used half-wave rectification of the secondary outputs of the power transformer. The result was that all of the devices drew power from the same half cycle of power on the data bus, resulting in asymmetric loading of the two half cycles. Future designs should use full-wave rectification to balance power usage between the half cycles of the AC power.

Data Bus Issues. The most serious issue with the data protocol chosen, the WorldFIP Fieldbus, was the basic architecture of the bus in which a bus arbitrator schedules data transmissions on the bus, and runs asynchronously with sampling of the sensors and use or production of the data by the engine control. This results in a substantially larger transport delay than would be required with a synchronous data transmission. A protocol such as MIL-STD-1553, in which transmission occurs under processor control, would eliminate this issue.

Another issue with the Fieldbus is the overhead associated with the protocol. In the Fieldbus protocol, a command-response exchange between the FADEC and a smart device has a minimum of 122 bits of overhead. In contrast, the MIL-STD-1553 protocol has about 35 bits of overhead. As a result of the increased overhead, the Fieldbus at 2.5 Mbps actually takes longer for a transaction than the MIL-STD-1553 at 1.0 Mbps (see Table 1). When the selection of Fieldbus was made in this program, it was believed that 5.0 Mbps Fieldbus would be available, but the higher speed parts have not materialized.

The software drivers provided with the WorldFIP products were not optimized for performance, therefore requiring excessive overhead when used in a real-time embedded control application. Optimization of the embedded software drivers would be required if this protocol continued in use for DCS.

The complexity of the Fieldbus protocol requires a large and complex communications co-processor (the FullFIP2 microcircuit) with its own private memory. This also resulted in an unacceptably long initialization process (about 500 msec) in which the code was loaded onto the FullFIP2 processor. The method for shortening the initialization time would be to add a private read-only memory for the FullFIP2, which would increase the size, cost and weight of an EIU.

Finally, there were some unexplained behaviors related to the Fieldbus. There was a high level of errors reported from the bus interface (typically, several errors per second). Generally these were handled automatically by the application software, but in some cases they did interfere with system operation. At one point, when investigating the source of these errors, an oscilloscope connected to the data bus appeared to show instances when two transmissions occurred on the data bus at the same time (these apparent dual transmissions were coincident with reported errors). No assistance could be obtained from the vendors of the Fieldbus interface to determine the cause of this problem, and no solution was found. While this may have been a result of some unknown inadequacy of the system cabling or configuration, it was a serious concern that could not be resolved.

A future program should investigate standard protocols for potential adoption and use in aerospace distributed control applications. A standard protocol would allow a common data link layer within the engine communication network. MIL-STD-1553 is strong candidate for use as a standard data bus, despite its relatively low data rate. This data bus has the advantages of low protocol overhead, simple implementation (two ICs), and widespread use within the aerospace industry. With a standard data packet definition, sensors and control components from various manufacturers would operate on the network without requiring a change in communication software design.

Future Efforts. It must remain the goal of future programs to develop Distributed Control Systems to the state where they will be ready for use on production engines. The following are a number of issues that remain to be addressed:

— Development of reliable and affordable high temperature electronics and packaging.
— Development of compact and rugged, flight-weight, engine-mountable packaging for EIUs.
— Development and selection of industry-standard communication data bus and power bus standards that can be implemented with available high temperature electronics, and have the data rate and attributes needed for future distributed engine controls.
— Demonstration that the projected cost and weight savings can be achieved via a distributed control system.
— Demonstration that the projected life-cycle cost savings and maintainability improvements can be achieved via a distributed control system.

While the challenges remain substantial, we believe that the rewards of distributed control systems for turbine engines make the effort to meet these challenges worthwhile.

Acknowledgments

This program was carried out by a team under US Air Force contract F33615-92-C-2220. Kenneth J. Roth developed the specifications for the Electronic Interface Units and the 100 kHz power supply, and coordinated their design and fabrication. AlliedSignal Aerospace Equipment Systems performed the detailed design of the EIUs at their expense, and carried out fabrication and design assurance testing of the EIUs. Lockheed Martin Control Systems conducted a preliminary design trade study on a partially distributed system. Honeywell Solid State Electronics Center supplied data on high temperature electronics on a no-cost basis. The GE custom data bus protocol for engine control applications was developed by General Electric Corporate Research & Development under an internally funded program.

References

Cegelec, 1994, "FullFIP2 Component User Reference Manual," 1st Issue, DPS 50262 a-en, Clamart, France.

ISA, 1992, "Fieldbus Standard for Use in Industrial Control System—Part 2: Physical Layer Specification and Service Definition," Instrument Society of America ISA-S50.02-1992, Research Triangle Park, NC.

UTE, 1990, "FIP bus for exchange of information between transmitters, actuators and programmable controllers, Data Link Layer," Union Technique de l'Electricité NF-C46-603, Courbevoie, France.