

# Langley Advanced Real-Time Simulation (ARTS) System

Daniel J. Crawford\* and Jeff I. Cleveland II†  
*NASA Langley Research Center, Hampton, Virginia*

A system of high-speed digital data networks was developed and installed to support real-time flight simulation at the NASA Langley Research Center. This system, unlike its predecessor, employs intelligence at each network node and uses distributed 10-V signal conversion equipment rather than centralized 100-V equipment. A network switch, which replaces an elaborate system of patch panels, allows the researcher to construct a customized network from the 25 available simulation sites by invoking a computer control statement. The intent of this paper is to provide a coherent functional description of the system. This development required many significant innovations to enhance performance and functionality such as the real-time clock, the network switch, and improvements to the CAMAC network to increase both distances to sites and data rates. The system has been successfully tested at a usable data rate of 24 M. The fiber optic lines allow distances of approximately 1.5 miles from switch to site. Unlike other local networks, CAMAC does not buffer data in blocks. Therefore, time delays in the network are kept below 10  $\mu$ s total. This system underwent months of testing and was put into full service in July 1987.

## Introduction

LANGLEY Research Center (LaRC) has employed flight simulation to support engineering research for at least 35 years. The vehicles most often studied are aircraft and spacecraft, but occasionally the facility is used to study more exotic systems such as trains, particle beams, flow control in wind tunnels, and aircraft landing carriages. The research engineer is usually testing a new or improved design in the areas of automatic or augmented control, handling qualities, guidance, navigation, flight management, terminal area air traffic management, air combat tactics, or some combination of these. The facility is not normally used for training. Different unrelated simulations are run simultaneously, and the same simulation equipment may be used in sequential 3-h periods throughout the day to support different independent studies.

Until the late 1960's, analog computers (continuous and parallel) were used to compute the system model because digital computers (discrete and serial) were not fast enough to keep up with the human pilot, i.e., fast enough to do real-time simulation. In 1967, LaRC installed two digital simulation systems, each composed of a central scientific mainframe computer (CDC 6600), a central signal conversion system, an analog signal distribution system (long-line twisted pairs), three simulation control consoles, an interval timer, supporting software, and various other peripherals. Unlike similar facilities, the mainframe was not dedicated to a single simulation. Up to three simulations, depending on resource requirements, could run together on a single computer, which would also do unrelated scientific processing in a background mode. This was accomplished by modifying the standard CDC SCOPE operating system. This approach had the advantage of keeping our facility current with a vendor-supported operating system. In 1976, we transitioned to the more

modern and interactive CDC NOS operating system and upgraded the mainframes to CDC CYBER 175's. Since then this system, with minor enhancements, has served LaRC well.

In 1981, we recognized two problems with the system and started looking for a solution. The first problem was that the 1967 equipment was aging rapidly and requiring more maintenance. Spares were difficult to acquire, and the distribution system cables were deteriorating. The second problem involved a basic change in the requirement. Digital avionic equipment and sophisticated digital cockpit display systems were becoming more and more a part of flight simulation. Integrating each type of these devices into the system required a difficult effort in both hardware and software. Simply replacing the equipment would be a very expensive solution to the first problem and would not address the second. The system requirement was to find or design and build a system which would perform as well as the current system, would be as functional, and would alleviate the digital device interface problem. It seemed apparent that the only direction to go was toward a high-speed local area network (or networks) employing distributed signal conversion equipment. Several flight simulation facilities that had already moved in this direction were visited. Certain difficulties became apparent from this survey. First, these systems are considerably slower than our previous system which could, for example, read in 80 analog-to-digital converters and 960 discrete inputs in 700  $\mu$ s after being triggered by the clock. This high I/O rate is extremely important since this rate, in conjunction with the speed of the central processor, determines the upper limits of the bandwidth and complexity of the simulations that can be run on the system. Second, the 50-Mbaud local networks are limited to distances of about 1,000 f, which is less than the distance to some of our simulator sites. And third, it was not apparent how networks could be completely reconfigured several times a day to accommodate the many combinations of simulator sites used by our research engineers.

During 1982 and 1983, a preliminary design was developed around a high-speed network technology called Computer Automated Measurement and Control (CAMAC). This network was developed and is widely used by people in the particle accelerator field. It has many laudable features, not the least of which is that it conforms to a national and international standard.<sup>1</sup> CAMAC as we use it has two major components: an addressable, powered, electronic chassis called a crate with a backplane bus called a dataway, and a ring master-slave net-

Received Feb. 19, 1987; revision received June 26, 1987.

Copyright © 1987 American Institute of Aeronautics and Astronautics, Inc. No copyright is asserted in the United States under Title 17, U.S. Code. The U.S. Government has a royalty-free license to exercise all rights under the copyright claimed herein for Governmental purposes. All other rights are reserved by the copyright owner.

\*Project Manager, ARTS System Project, Computer Systems Branch, Analysis and Computation Division.

†Software Project Manager, ARTS System Project, Computer Systems Branch, Analysis and Computation Division.

work called a highway, which connects crates and the mainframe.

The highway has a clock rate of five bytes per  $\mu\text{s}$  and a maximum usable data rate of three bytes per  $\mu\text{s}$ . However, it uses very little handshaking (protocol), it does not buffer data in blocks within the network, and it does not have to compete (bus connection) for network services. Therefore, for our application, it is superior to the 50-Mbaud local networks used for file transfers. By developing or enhancing certain features, we believed its effective data rate would almost double that of our present system and that it could achieve distances beyond our requirement. It also appeared that a switch could be built to automatically configure networks under the control of a mainframe computer program. And finally, the task of interfacing digital devices to the mainframe was reduced to developing a printed circuit card (module) for the crate with a very straightforward and well-documented interface to the crate bus (dataway).

This design, a plan, a projected budget, and a schedule were all presented to LaRC management in late 1983, and later received a go-ahead for implementation. Since then, the system has been implemented and is currently being used for research. The intent of this paper is to provide a coherent functional description of the new LaRC Advanced Real-Time Simulation (ARTS) System.

### Highways

The LaRC Advanced Real-Time Simulation System employs eight high-speed digital networks called CAMAC highways. At any given time, up to six totally independent simulations can be accommodated simultaneously. An aircraft simulation model is solved on one of two mainframe computers, and the simulation is normally assigned one highway. In certain special cases, a second highway can be assigned to a job. The purpose of the network is to communicate data between the central computers and the simulation sites (control console, cockpit, display generator, etc.). At setup time, each job requests the sites it needs by computer control statement. If the sites are available, then the network is dynamically configured, and the job will be elevated to real-time status.

Figure 1 is a block diagram of a single highway consisting of three sites. The configuration switch is cabled to 25 sites (ex-

pandable to 44) and can accommodate up to 12 highways. This switch will be discussed further in a later section. The elements of the CAMAC highway are the Block Transfer Serial Highway Driver (BTSHD), the Fiber Optic U-Port Adaptor (FOUPA), the Block Transfer Serial Crate Controller (BTSCC), the List Sequencer Module (LSM), and the CAMAC crate. The block diagram has several shortcomings that should be pointed out. The diagram simplifies the switch matrix by showing only one network and portrays the FOUPA's as stand-alone devices, whereas in reality they are CAMAC modules deriving power from the dataway. Furthermore, the diagram does not show either the BTSCC module, which occupies the two rightmost slots of every crate, or the LSM module, which is usually adjacent to the BTSCC. Figure 2 is a schematic representation of the contents of a crate. The figure shows the various CAMAC modules that might be found in a typical site crate.

Three features of the networks were developed to meet the LaRC requirement. First, the mainframe computer (CDC CYBER) interface to the BTSHD was developed. Second, the block transfer capability was developed to meet our performance requirements. This capability resides in the BTSHD, the BTSCC, and the LSM. Finally, the FOUPA was developed to satisfy our site distance problem. The sites are from 350–6,000 f from the switch.

Prior to these developments, a CAMAC message was approximately 19 bytes long, including crate address, module address, function, one 24-bit CAMAC data word, a Cyclic Redundancy Check (CRC), and a response. The addition of the block transfer capability allowed for the inclusion of many CAMAC data words in a single message. During block transfers, data reads or writes proceed synchronously at one 24-bit CAMAC data word per  $\mu\text{s}$ . This is three to four times faster than the single-word message rate. A good description of the block transfer mechanism and its performance as a function of block size is given in Ref. 2. In order to support this data rate, the byte serial highway is clocked at 5 bytes per  $\mu\text{s}$ , and the bit serial FOUPA proceeds at 50 bits per  $\mu\text{s}$ .

Besides the CAMAC standard message, there are two modes of block transfer. In the first, the entire block goes to a single module within a crate. This is typically used in a mainframe-to-site-computer block transfer. It is implemented

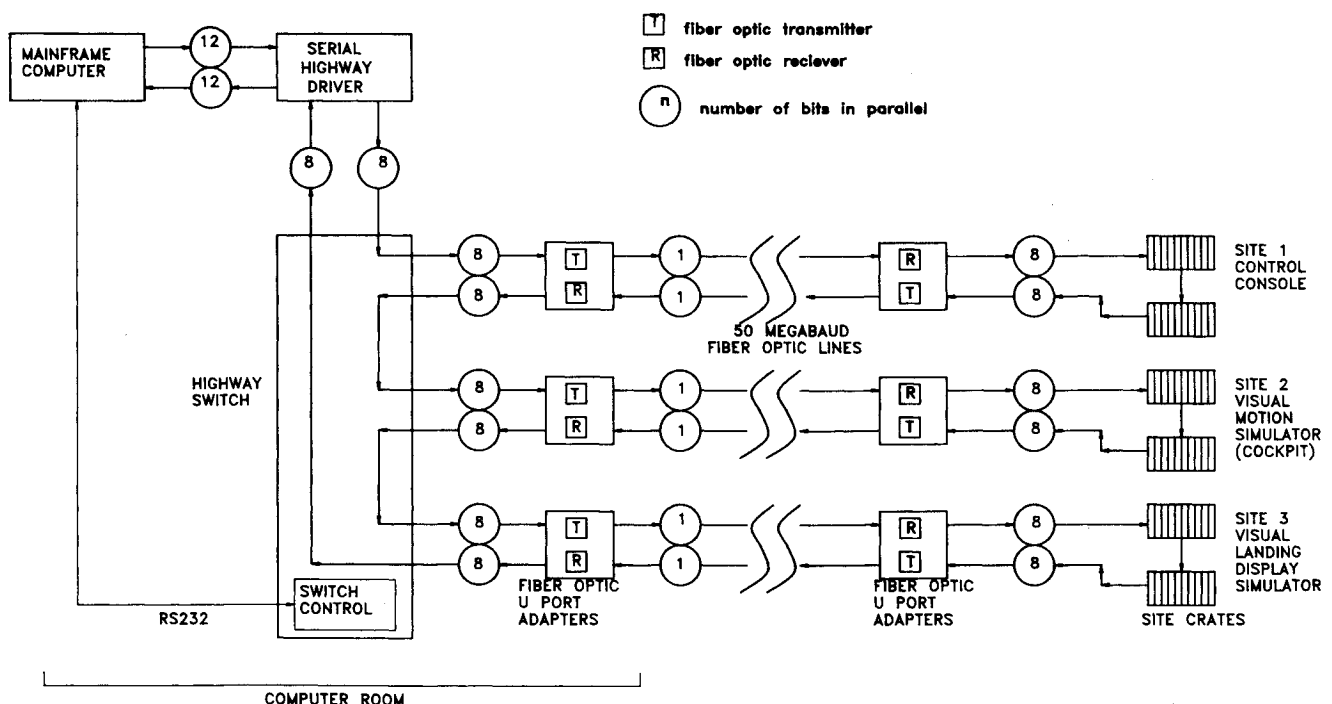


Fig. 1 Typical simulation network.

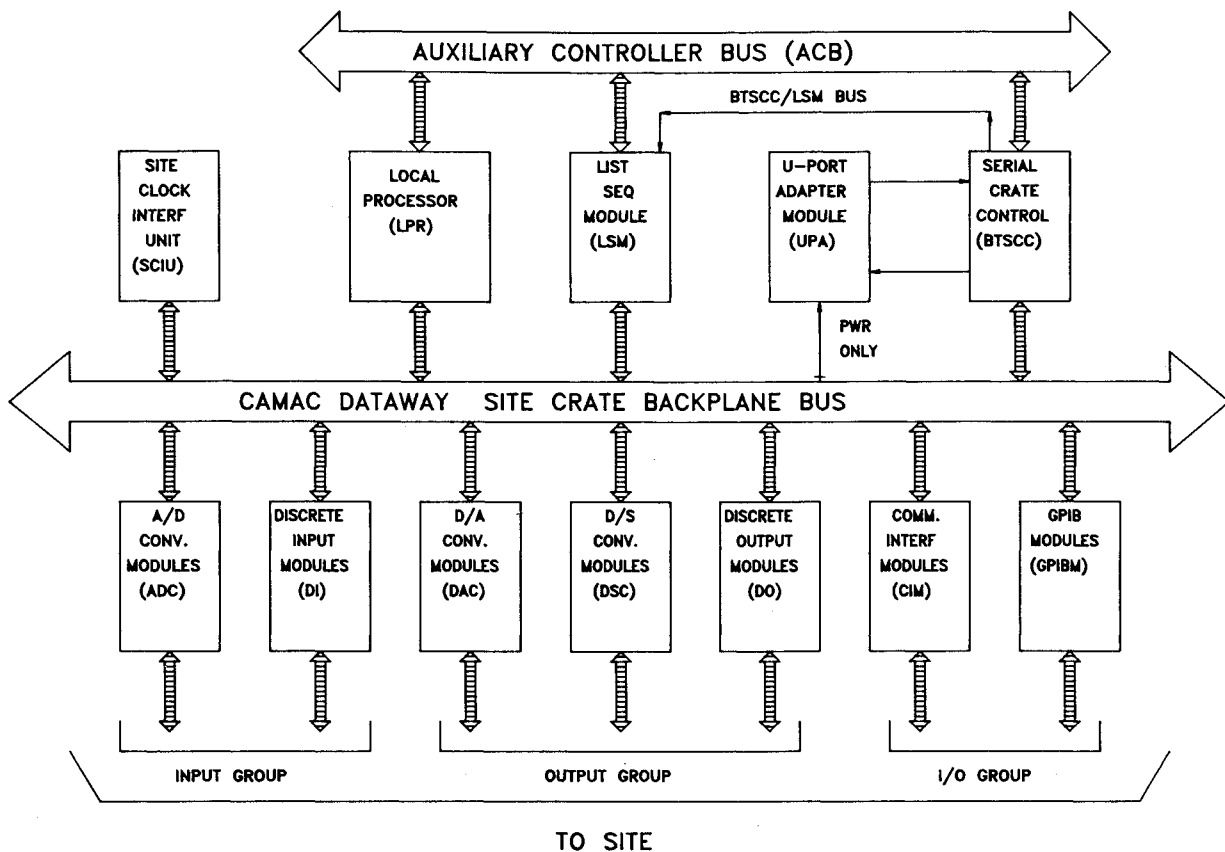


Fig. 2 CAMAC crate contents.

by the BTSCC repeating the module-select and function bits on the dataway for each CAMAC word (crate cycle). In the second block transfer mode, the block, either on read or write, is divided among several modules within a crate. This mode is typically used to combine all the input converters into a single block for transfer to the mainframe or to disperse a block from the mainframe to all the output converters in a crate. It employs the LSM module, which is loaded by the mainframe computer at setup time with up to four lists of module-select and function bits. When this type of block transfer is in progress, the BTSCC acquires module number, function, and subaddress for each sequential CAMAC word in the block from the indicated list in the LSM. These are placed on the appropriate dataway lines while the data word is being read or written.

The data channels on the LaRC mainframe computers have a maximum block transfer rate of 12 data bits per 500 ns. The crate cycle is  $1 \mu\text{s}$ , and the dataway has 24 data lines each way. There is no block buffering in the BTSHD or BTSCC. Therefore, in general, data flows from the signal converters to the mainframe channel memory (and vice versa) at the 24-Mbaud rate. The highway is a ring network, so messages travel through all assigned crates. The time it takes for a block leaving the mainframe to reach converter modules 1 mile away is in the order of  $10 \mu\text{s}$ . This includes delays from cable length, switch matrix, and passage through nontargeted crates.

The BTSHD is highway master, and all communication, with one exception, consists of the driver functioning, writing, or reading a crate module. The one exception is the generation of a demand message by the BTSCC in response to a Look-At-Me (LAM) being set by one of its crate modules. A set of L lines (LAMs) on the dataway connect each crate module (point to point) to the crate controller. When a module activates its L line, the BTSCC sends a message (demand) to the BTSHD. The message contains the crate address and five additional user-defined bits of information. This information is then passed on to the mainframe computer channel driver and in

some cases to the simulation program. The ARTS system uses LAM's in three places. First, the site clock interface unit generates a demand message when it determines a frame compare. This process is discussed further in the following clock section. The frame compare starts the cycle of input-compute-output in the central computers. The frame time is typically 20–40 m but can vary from 5–1,000 m. Second, the LAM is an important element in the protocol between site computers and central computers. Third, a button on the control console labeled Program Interrupt will generate a demand message. When a demand message is triggered from this source, the central computer channel driver interrupts the simulation job and brings the job into an error processing mode. It would normally be used during the program development phase if the analyst requires an abnormal interrupt of the simulation.

The periodic real-time frame proceeds, with some variation, as follows. Upon sensing a frame compare, input signal conversion commences in appropriate modules at all sites assigned to the job. Simultaneously, output data is converted and a frame compare demand message is sent to the mainframe. The CYBER channel driver program responds to this message by reading input data from appropriate crates. In almost all cases, this consists of only one block per crate, thanks to the LSM. This process is table-driven, and the data is stored in the mainframe central memory. When input is complete, the job becomes a candidate for the central processor. The channel driver services a lower priority queue of highway communication requests. The central processor advances the solution of the equations of motion one time step and signals that output is ready. The channel driver then writes the data via the network to the output converter module buffers and puts itself into a polling loop awaiting the next clock compare.

Features of the highway system are summarized in Table 1. The system was designed and built to LaRC requirements by Kinetic Systems Corporation (KSC) of Lockport, Illinois.<sup>3</sup> The performance specification of the highway system is given in Table 2. Since early 1987, the highway system has been per-

forming very well, supporting flight simulation at the required data transfer rates.

### Network Switch System

The purpose of the network switch system is to provide complete connectability between the simulation applications on the mainframe and the various simulation sites. Upon

**Table 1 Highway system features**

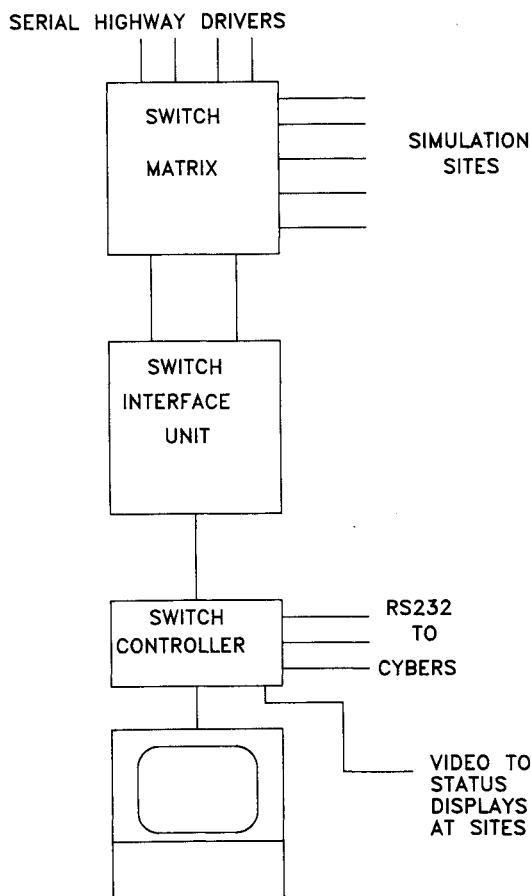
Conforms to international standards with extensions
Single-master ring network
Module generated demand message to master
Dynamically reconfigurable through network switch <sup>a</sup>
CYBER mainframe interface <sup>a</sup>
CRC and transverse parity
Mini/microcomputer interface <sup>a</sup>
Single-module block transfer <sup>a</sup>
Block transfer fan in/fan out with list mode <sup>a</sup>
No block buffering within network

<sup>a</sup>Developed to LaRC specification.

**Table 2 Highway system specifications**

Byte serial electronic transmission at 5 Mbytes/s
Bit serial fiber optic transmission at 50 Mbits/s <sup>a</sup>
Effective data rate of 24 Mbits/s <sup>a</sup>
Distance greater than 6000 f <sup>a</sup>
More than 20 available slots/crate
Six expandable to nine simultaneous simulations <sup>a</sup>
Eight expandable to 12 highways <sup>a</sup>
Two expandable to three CYBER mainframes <sup>a</sup>
25 expandable to 44 sites <sup>a</sup>
Up to 62 crates/network
7 $\mu$ s/mile fiber optic transport delay
100 ns crate pass-through

<sup>a</sup>Developed to LaRC specification.



**Fig. 3 Network switch block diagram.**

request, any sensible combination of available sites can be combined into a local computer network in support of a single simulation. This network configuration for a given simulation is done during the initialization phase, after a highway has been assigned by the scheduling software. The application job requests sites by resource request statements, and if the sites are available, the switch system will electrically and logically configure the network without disturbing other running simulations.

In its initial configuration, the LaRC ARTS system will have four CAMAC serial highways on each of two mainframe computers, and 25 simulator sites. The network switch, however, is built for a maximum of 12 highways and 44 sites. Each computer will be capable of running three independent simulations simultaneously with the capability of assigning a second highway to any simulation. For connection to the mainframes, each highway requires a computer channel and a BTSMD. The network switch, the central clock, the BTSMD's and a FOUPA for each site are located close to the central computers. Located in an adjacent attached building are 16 simulation sites, which include the control consoles and are at distances from 300–500 f from the switch. In addition, nine sites are located in five remote buildings, which are at distances from 1,000 f to slightly over a mile.

The switch system has four major subsystems: the switch matrix, the controller, the matrix interface unit, and the display system. Figure 3 is a block diagram of the switch system.

The switch matrix performs the actual highway/site switching. The switch matrix chassis has slots for two input boards, two output boards, and 11 pairs of site boards. One pair of site boards services four sites. Each highway passes from the BTSMD through all the site boards and then returns (ring network) to the BTSMD. Each highway is latched to connect to a site or to bypass the site, but the design is such that a site can be connected to only one highway at a time. Each of these highway/site crosspoints involves eight parallel signal bits and a 5-MH clock bit. When a site is connected, the data flows on the highway out to the site and then is returned to the same site board. The data then proceeds to the next highway/site crosspoint.

The interface unit provides physical control and monitoring of the matrix. When in its local mode, the matrix interface unit is controlled from its front panel. The unit has two distinct functions. The first is to monitor by Light Emitting Diodes (LED's) the nine-bit streams at any single crosspoint. Thumbwheel switches select the highway/site crosspoint to be displayed. Two additional latched LED's indicate parity error or loss of highway signal clocking. This diagnostic capability is expected to be enhanced in the future. The second function is to control and monitor the state of the matrix. By a second set of thumbwheel switches, a site can be selected and its connected highway is displayed. In addition, one can select a crosspoint and connect or disconnect the highway from the site. Normally the matrix interface unit will be in the remote mode, and these monitoring and control functions are passed through a parallel port to the switch controller. However, at this writing, the data stream monitoring has not been implemented at the controller level.

The switch controller provides high-level control of the switch by either the mainframe or a local terminal. The switch controller is an Intel 8085 single-board computer with a video board and a communication board. It has a read-only program memory and a RAM with a battery backup for table storage. There are no disks or other secondary storage. The controller tables contain the state of the system: which jobs are running using which sites, which sites are free, and which are not available and why. This information is passed upon request to any job on either mainframe by way of the communication board. It is also distributed as a display through a network of video monitors to various sites and offices. All sites and jobs have mnemonic identifiers such as VMS for the Visual Motion Simulator. The tables are maintained

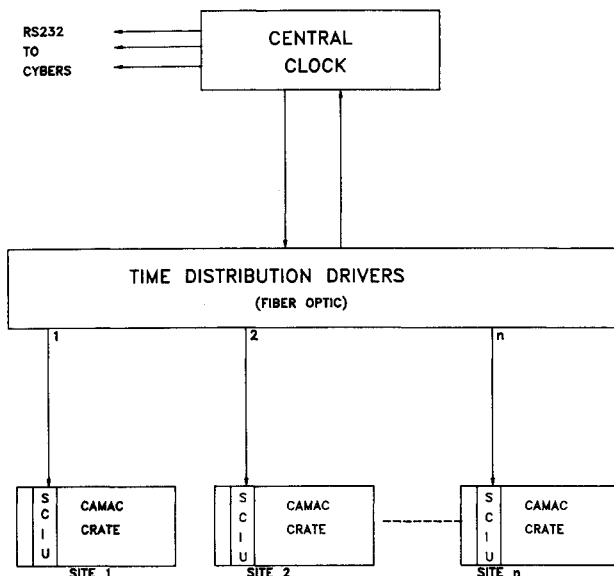


Fig. 4 Clock system block diagram.

Table 3 Network switch features

Complete dynamic connectivity
12 highways by 44 sites matrix
Less than 2 $\mu$ s switch pass-through delay
Highway signal regenerated on each site board
Control and status through CYBER
4 RS-232 controller ports
Private configuration status video display network
Alternative off-line control
Diagnostic monitoring of signal quality and configuration state

dynamically as sites are requested by the mainframe computers and are assigned to jobs (connected to highways).

The switch controller has four communication ports, and the software is written to accommodate four computers. In addition to the two simulation computers, a small computer used for software development is connected to the third controller port. The fourth port will be used by a microcomputer driving an off-line diagnostic highway currently being developed.

When a simulation job requests real-time scheduling, the software requests a highway channel and is assigned any free channel. The scheduling process then searches its files for the job's associated table of sites. The scheduler communicates with the switch controller by way of the standard interactive facility (RS-232) and requests that the necessary sites be connected to the assigned highway. If the sites are available, they are assigned physically through the matrix interface unit and the switch controller updates its status tables and displays. From that point on, the network is configured and the switch is invisible to the simulation job. If, however, the sites are unavailable, the job is refused real-time status and the user is informed of the site conflict. At this point, the user can either consult a status monitor to find out why a site is unavailable or he can have switch status displayed at his terminal by an appropriate request to the job scheduler.

Features of the network switch system are summarized in Table 3. The switch matrix was built to our requirement by Kinetic Systems Corporation. The other components were built in-house.

We have had the system in operation for over a year. The switch system meets our requirements and is both stable and reliable.

### Clock

The purpose of the clock system is to synchronize simulations to the real-time clock and to other simulations. The clock

system is composed of a central unit and multiple Site Clock Interface Units (SCIUs) connected by a separate fiber optic star network. A block diagram of the clock system is presented in Fig. 4. Two distinct time intervals are broadcast by the central unit to each site on a single fiber. The first time interval has a constant 500- $\mu$ s period. The frame tic count is set in the SCIU by scheduling software and is decremented once for each occurrence of this time period. When the count reaches zero, each SCIU issues a job-specific periodic signal called "frame compare" that is the beginning of a real-time frame. The frame time is determined independently for each job but, of course, must be a multiple of 500  $\mu$ s. The second clock signal, called the "job sync tic," has a longer period, called the Clock Common Multiple (CCM), which is set by thumbwheel switches on the central unit to a multiple of 500  $\mu$ s, from 1-65,535. This longer interval is used to synchronize jobs and is used by the static scheduler (software) in its time scheduling calculations. In order to enter real-time status, a job's requested frame time must divide (without remainder) into this CCM. In addition, the SCIU will wait to issue the first frame compare (of the session) simultaneously with the occurrence of the job sync tic. These two constraints result in the condition that all jobs in the system, regardless of frame time, receive a frame compare simultaneously with each occurrence of the job sync tic. This is used for job synchronization.

The central unit employs an accurate temperature-controlled 5-MHz oscillator from which it derives the two time interval signals. It combines the two intervals into a single 1-MHz signal and uses a daisy chain of fiber optic transmitters to send the signal out to the sites. This signal is sent via a 100- $\mu$ m fiber that shares a cable with the CAMAC highway. The central clock also employs an Intel 80/24 single-board computer and an Intel 534 communication board for RS-232 communication with the mainframe computers. The communications between the mainframes (up to three) and the central clock consists of the clock supplying its status registers upon request. These registers contain the value of the CCM and various hardware status bits. This information is used by the static scheduler.

At every site there is one SCIU. The SCIU is a CAMAC module residing in one of the site crates. It has a fiber optic receiver and it decouples the two time intervals transmitted from the central clock. The SCIU is preset by way of CAMAC highway functions at job scheduling time. One of its registers is set to the job frame tic count. At all but one site for a given simulation, the only function of the SCIU is to count down the central clock and issue a trigger pulse to start the signal conversion process in all the relevant CAMAC modules. This clock frame compare signal is brought to an internal line on the crate bus (dataway) through the front panel of the LSM. The LSM has provisions to route this frame compare signal to the other crates at the site. The SCIU maintains a status register that can be read via CAMAC function by the mainframe computer. It will report nonsynchronization, i.e., the occurrence of a job sync tic without a frame compare.

At one site for each simulation, usually the control console, the SCIU is preset for an additional function. On each frame compare the SCIU initiates a demand message that is sent up the highway to the mainframe where it starts the real-time frame process (input-compute-output). This SCIU LAM must be reset every frame by reading the module's status register.

Features of the clock system are summarized in Table 4. Since the clock system is recognized as a single point of failure, it has been built with self-monitoring features and can be quickly repaired by board swapping. Several interesting quality assurance features of the central clock deserve being noted. First, the clock signal directed to the sites is daisy-chained through all of the fiber optic transmitter boards and then returned to the clock. Any interruption of the signal will cause a status bit to be set and the display on the front panel to start flashing the status (a Status Alert). Second, the clock oscillator is continually compared to the National Bureau of Standards radio station WWVB signal over a 1,000 s period. This

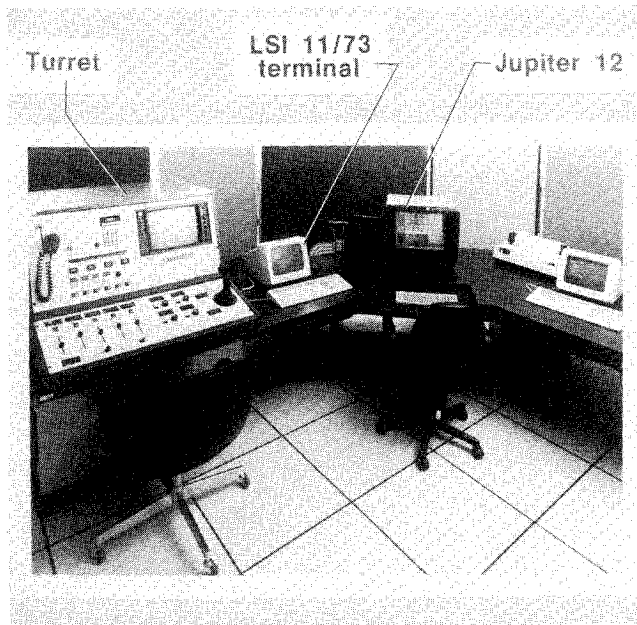


Fig. 5 Console components.

Table 4 Clock system features

---

Single time base for all simulations  
 Temperature-controlled oscillator with excellent aging characteristics  
 Interjob synchronization  
 Self-monitoring and trouble reporting  
 Three-mainframe access (RS-232) for status and reporting  
 Allowable site distance and number of sites far exceed requirement  
 Fiber optic star network shares cable with CAMAC highway  
 One site unit services multiple crates  
 Triggers start of conversion and alerts mainframe by highway message each time it counts down to a frame compare

---

difference is periodically monitored to detect any unexpected drift. Third, both the oscillator signal and the 500- $\mu$ s tic are continuously monitored. The interruption of either will trigger a Status Alert. Finally, the system has redundant interval counters that are continuously compared to the primary pair. A brief physical description of the clock system is given in Table 5.

In summary, the frame time for each job is a multiple of 500  $\mu$ s and must divide without remainder into the CCM. At job scheduling time, this frame period is set into the SCIU at each site assigned to the job. All the SCIU's initiate signal conversion at their respective sites when they count down the central short period clock to a frame compare. An example of this conversion is the sampling and conversion of analog inputs. One of the SCIU's initiates a message up the highway, starting the frame process on the large central computer. Direct communication (RS-232) between the central clock and the job running on the mainframe computer is relatively slow. It occurs at job scheduling time and during periodic quality assurance polls.

The clock system has been in service since Nov. 1985. All features have been successfully tested, and we have had no reliability problems. Oscillator accumulated time error as measured against WWVB exceeds our requirement for accuracy.

### Signal Conversion Equipment

Three types of output converter modules (digital-to-analog, discrete output, and digital-to-synchro) and two types of input converter modules (analog-to-digital and discrete input) were designed and built to LaRC specification in support of the ARTS system. In addition, two off-the-shelf modules, Gen-

Table 5 Clock system physical characteristics

#### Central clock

---

Standard Multibus chassis (19-in. rack mount)  
 Four Multibus boards  
 Commercial  
 Intel 80/24 single-board computer  
 Intel 534 communication board  
 Special in-house developed  
 Interval generator  
 5-MHz oscillator board  
 Front panel controls/displays  
 Power on/off  
 Reset  
 Five-digit thumbwheel switches (CCM)  
 Five-digit HEX LED display  
 Normal — CCM value  
 Fault — flashing error code

---

#### Site Clock Interface Unit

Single-width CAMAC module (special in-house developed)  
 Front panel fiber optic input from central clock  
 Front panel output for frame compare  
 Optional jumper for frame compare output on CAMAC P1 line  
 Front panel LED indicators  
 N — card addressed  
 ENBL — frame clock enable  
 RUN — frame clock running  
 FRAME — frame compare  
 CLK — tic  
 SYNC — job sync  
 ERROR — job sync error

---

#### Fiber optic distribution system

Single-width CAMAC card with eight front-panel-mounted fiber optic transmitters  
 Rear-panel Lemo connectors for encoded clock in and out powered by standard CAMAC crate  
 Site expansion by eight with each additional transmitter card

---

eral Purpose Interface Bus (GPIB) and RS-232, were acquired. The converters are high-quality and commercially available. Their features are described in greater detail in Ref. 4. All these converter modules meet or exceed our expectations and specifications.

### Site Computer Interface

The purpose of the interface is to provide a high-performance data communication path between the central computers and the site computers. Initially, interfacing was done with a CAMAC RAM module that could be read and written both by the central computers over the highway and by the site computers through the crate dataway bus. To overcome deficiencies with this approach, two interfaces have been developed in-house and are currently being tested. These interfaces support two DEC bus structures, the Q-bus and Unibus. Additional details on these interfaces is found in Ref. 4.

### Console

One of the sites on a highway is the simulation console. The console is used by the simulation programmer, simulation operator, and the researcher to control and monitor the simulation application. The console (Fig. 5) consists of four major components: a Jupiter 12 high-resolution 19-in. color graphics system used as the main control element, two CAMAC crates containing an LSI 11/73 used to communicate with the network, a turret assembly used for pilot-like analog input/output, and the console cabinet. The Jupiter 12 monitor is overlaid with a touch-sensitive screen to provide additional inputs. The Jupiter 12 is connected to the console LSI 11/73 via RS-232 and high-speed Direct Memory Access (DMA). The turret assembly contains a Prolog microprocessor that is connected via RS-232 to the console LSI 11/73. These data paths are illustrated in Fig. 6. Additional console information is found in Ref. 4.

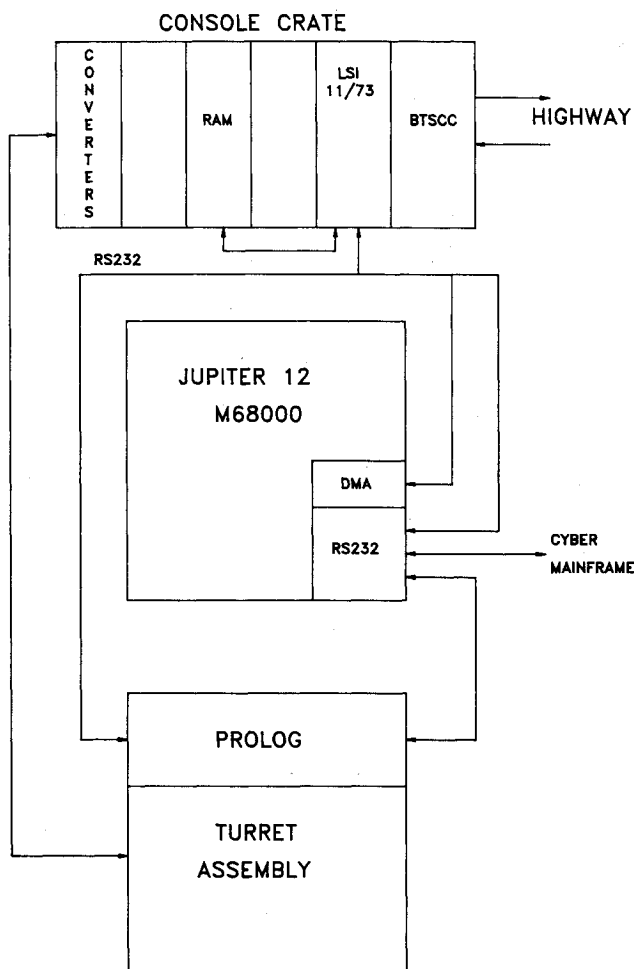


Fig. 6 Console block diagram.

### Central Computer Software

The development of the system software is an effort that is equal in magnitude to that of developing the hardware. However, for several reasons including space considerations and its lack of generality, it will not be discussed here. A general description of the software can be found in Ref. 4, and detailed documentation is currently being produced.

### Diagnostics and Maintenance

The ARTS system is expected to be in service for at least ten years. It will be maintained under an on-site service contract and will require engineering and computer analyst support throughout its lifetime. The emphasis for diagnostic and quality assurance testing has been to test off-line wherever and whenever possible. Major elements were designed to be as self-diagnosing as possible, within the constraint of time-critical operations. Since all sites contain local intelligence, testing can be done off-line with the local processors. In addition, the central software does diagnostic testing during the initialization phase when time-critical operations are not required. Further information on diagnostics and maintenance functions can be found in Ref. 4.

### Concluding Remarks

The need for a new real-time flight simulation system for the NASA Langley Research Center was recognized in 1981. During the following 18 months, the requirements for a new system were formalized and a preliminary design was established. The design included signal conversion, signal distribution, clocking, new control consoles, and supporting software. The design differed radically from the system in place in that it had 10-V as opposed to 100-V reference, and converters were distributed to the simulator sites rather than being centralized.

The signal distribution system was serial digital rather than parallel analog. In addition, each site would have either a mini or microcomputer, and in the case of the control console, a three-microcomputer network would be employed. The central point of the design was a system of high-performance local area CAMAC networks that could be dynamically configured into almost any arbitrary combinations of simulator sites under the control of the flight simulation application job. This design included significant enhancements to the CAMAC network, which is used extensively in the nuclear accelerator field for data acquisition and control. The great majority of the system hardware was not commercially available and was designed and built to specification.

This preliminary design, along with a budget, a schedule, and a development and implementation plan, were presented to and approved by LaRC management in late 1983. Since then, an in-house team of analysts and engineers have been employed in the final design, development, and testing of the system. Kinetic Systems Corporation of Lockport, Illinois, was the principal vendor, and they designed and developed the CAMAC networks, the network switch matrix, and the converter equipment to specification. Other elements such as the control console, clock system, switch control and interface, site computer interface, maintenance and diagnostic tools, and all software are being or have been designed and developed in-house.

The first phase of the development and implementation plan was to develop hardware prototypes and demonstrate proof of concept. This phase was successfully completed in Dec. 1985, three months ahead of schedule. At that time, the prototype highway, clock, switch, console, and a simulator cockpit were combined to do a piloted aircraft simulation. The software was primitive but its hardware interface components were quite serviceable, since they had been used to test the hardware prototypes. The aircraft was flown from the simulator cockpit, and the simulation was controlled from the touch screen of the console graphics system. The impressive capability to read and write blocks of converters at the rate of three every 2  $\mu$ s with transport delays of less than 10  $\mu$ s between mainframe and site was clearly demonstrated.

Following that successful test, the project entered its second phase, which was concerned with the development, integration, and testing of the supporting software and development and installation of the production hardware. The end of the second phase was marked by a milestone known as "the three-job demonstration." This required that three piloted research simulations be run simultaneously on the two CYBER 175 computers while unrelated batch jobs were running in the background. The operating system used was a modified version of the current NOS release, which was running on other (nonreal-time) computers in our facility. Application programs were converted to FORTRAN 77 and integrated into the new system. Seven simulation sites, including three consoles, were used in the test. We succeeded in passing this milestone in April 1987. Since then the system was subjected to user confidence testing and system software quality assurance testing. It passed all tests and went into full service in July 1987. The period since the ARTS system went into service has been used to support the system and enhance it by responding to user's comments and criticisms. Some of the final system hardware will not be acquired until 1988. The ARTS system represents a significant advance in the technology supporting real-time flight simulation.

### Acknowledgment

The work described in this paper was accomplished by a team of LaRC in-house engineers and analysts from the Computer Systems Branch and the Analysis and Simulation Branch. The team included NASA personnel, support contractors from the Unisys Corporation (formerly System Development Corporation and Sperry Corporation), and one analyst from Control Data Corporation. The network equipment,

including the switch matrix and the signal conversion equipment were designed and built by Kinetic Systems Corporation. Their efforts were critical to the success achieved.

### References

<sup>1</sup>ANSI/IEEE Standards 583, 595, and 675, Institute of Electrical and Electronic Engineers, 1976.

<sup>2</sup>Cleary, R. T., "Enhanced CAMAC Serial Highway System," presented at the IEEE Nuclear Science Symposium, San Francisco, CA, Oct. 23-25, 1985.

<sup>3</sup>Cleary, R. T., "An Ultra-High-Speed CAMAC Interface to a Large Flight Simulator System," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-22, No. 5, Sept. 1986, pp. 618-627.

<sup>4</sup>Crawford, D. J. and Cleveland, J. I., "The New Langley Research Center Advanced Real-Time Simulation (ARTS) System," AIAA Paper 86-2680, Oct. 1986.

*From the AIAA Progress in Astronautics and Aeronautics Series...*

## **COMBUSTION DIAGNOSTICS BY NONINTRUSIVE METHODS - v. 92**

*Edited by T.D. McCay, NASA Marshall Space Flight Center  
and*

*J.A. Roux, The University of Mississippi*

This recent Progress Series volume, treating combustion diagnostics by nonintrusive spectroscopic methods, focuses on current research and techniques finding broad acceptance as standard tools within the combustion and thermophysics research communities. This book gives a solid exposition of the state-of-the-art of two basic techniques—coherent antistokes Raman scattering (CARS) and laser-induced fluorescence (LIF)—and illustrates diagnostic capabilities in two application areas, particle and combustion diagnostics—the goals being to correctly diagnose gas and particle properties in the flowfields of interest. The need to develop nonintrusive techniques is apparent for all flow regimes, but it becomes of particular concern for the subsonic combustion flows so often of interest in thermophysics research. The volume contains scientific descriptions of the methods for making such measurements, primarily of gas temperature and pressure and particle size.

*Published in 1984, 347 pp., 6 × 9, illus., \$39.95 Mem., \$69.50 List; ISBN 0-915928-86-8*

TO ORDER WRITE: Publications Dept., AIAA, 370 L'Enfant Promenade, SW, Washington, DC 20024