NASA Technical Memorandum 83231

Design and Implementation of a Telecommunication Interface for the TAATM/TCV Real-Time Experiment

Joseph D. Nolan

DECEMBER 1981

NASA

NASA Technical Memorandum 83231

Design and Implementation of a Telecommunication Interface for the TAATM/TCV Real-Time Experiment

Joseph D. Nolan
Langley Research Center
Hampton, Virginia

National Aeronautics
and Space Administration

Scientific and Technical
Information Branch

1981

SUMMARY

The traffic situation display experiment of the Terminal Configured Vehicle (TCV) research program, in conjunction with the Wallops Flight Center, requires the integration of NASA's Boeing 737 aircraft, in the form of a live radar target, into a terminal area that is being simulated on a Control Data CYBER 175 computer. The Terminal Area Air Traffic Model (TAATM) real-time simulation program, operating on a CYBER 175 at Langley Research Center, communicates with the Wallops computer facility via a 4800-baud bidirectional data link (tie-line). Tie-line data from the TAATM simulation program are processed by the Wallops computer facility which generates and then transmits the airborne traffic situation displays to the NASA aircraft. Input data to TAATM and output data from TAATM are updated every 4 seconds (radar sweep rate) via the tie-line.

This report describes the tie-line communications interface, including the protocol, as implemented in a real-time environment on the CYBER 175 computer. Special emphasis is given to the design features incorporated in the CYBER communication software, which provide maximum control and flexibility to the TAATM simulation program in its use of the tie-line.

INTRODUCTION

An aircraft research program at Wallops Flight Center, coupled with the Terminal Configured Vehicle (TCV) research program, necessitated the design and implementation of a time-critical telecommunications data link between the Wallops facility and the Langley Central Scientific Computer Complex. The principal application of this link is to transfer data between the TCV aircraft flying at Wallops and the Terminal Area Air Traffic Model (TAATM) real-time simulation program at Langley Research Center.

The TAATM program provides a comprehensive simulation of up to 40 aircraft operating in a terminal area with a radius of 40 n. mi., and was designed to allow research studies of improved methods for traffic management. The target aircraft respond, with preset statistical variances, to controller commands and are displayed on a graphics console, similar to current air traffic control (ATC) radar displays, at a 4-second update (sweep) rate. The objectives of the combined TAATM/TCV experiment were

1. To provide a real aircraft and crew response in addition to the simulated aircraft (computer-generated targets) in TAATM

2. To allow investigation of flight-crew roles in the traffic-management task through the provision of an up-link display of other aircraft in the vicinity of a controlled aircraft

The requirements discussed in this report were imposed by the airborne traffic situation display (TSD) experiments of the TCV program in conjunction with the Wallops Flight Center (fig. 1). This facility utilizes a Honeywell 716 computer as a real-time data handling system. The TCV aircraft is a Boeing 737 equipped with advanced guidance displays and automatic navigation, guidance, and control systems. Using a bidirectional data link, the TSD experiments integrate the TCV aircraft that

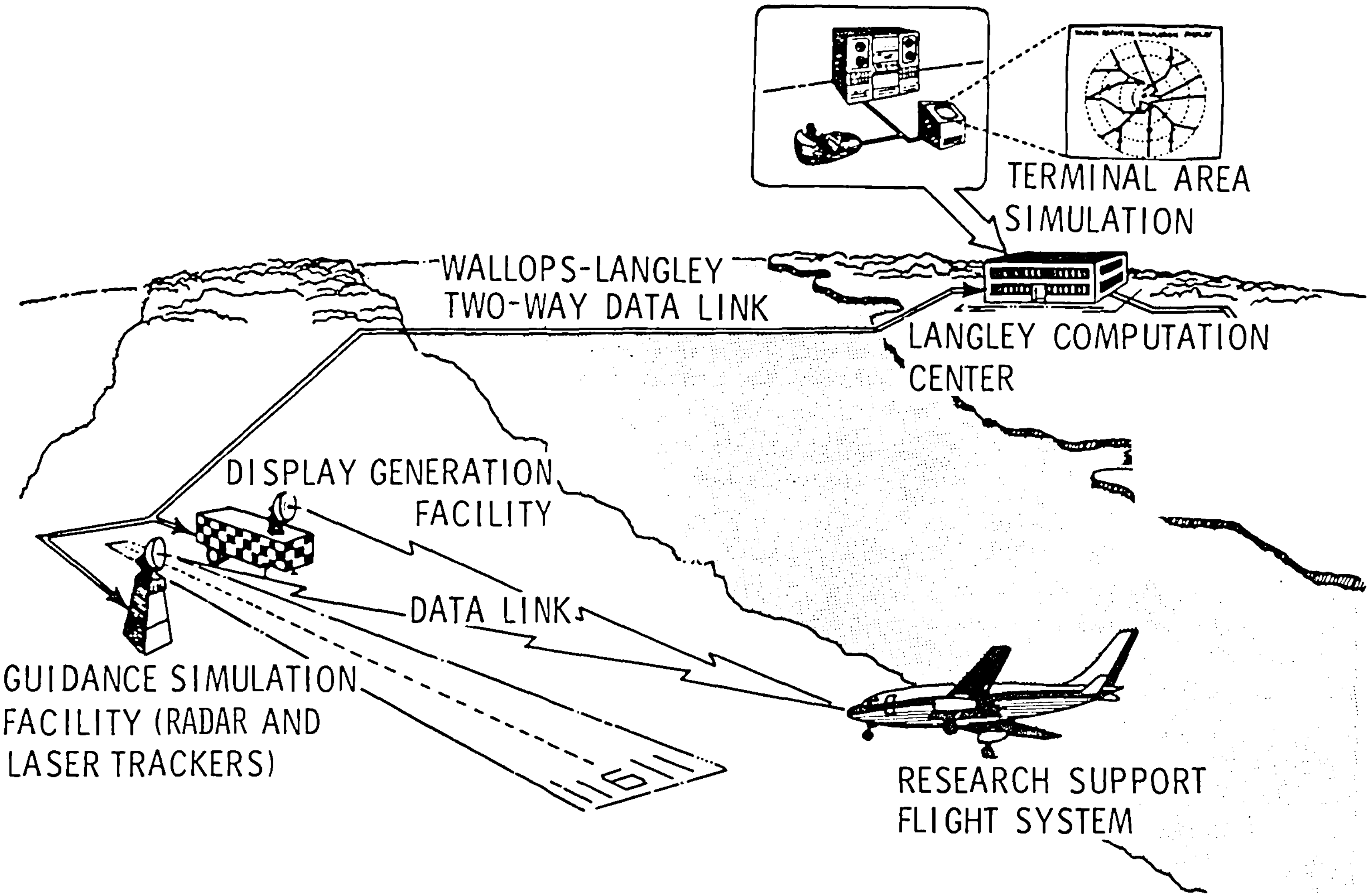


Figure 1.- Wallops-Langley aircraft research facilities.

is being tracked by the radar at the Wallops facility with a simulation of an airport terminal area being generated by the TAATM program in a real-time environment on Langley's Control Data CYBER 175 computer.

The Control Data CYBER 175 computer is one of two computer systems that process the real-time simulation applications at the Langley Central Scientific Computer Complex. Appendix A describes the real-time simulation subsystem as integrated into the computer complex. Figure 2 presents an overview of the tie-line system components and data paths.

TIE-LINE DATA AND TIMING REQUIREMENTS

The integration of the TCV aircraft into a terminal-area environment is a major objective of the traffic situation display (TSD) experiment. To help achieve this objective, a facility is needed at Wallops to coordinate the required input/output data for the display experiment. The Honeywell 716 computer (HW 716) functions as the data coordinator.

The HW 716 interfaces the tie-line to three input/output sources: the Honeywell 316 computer (HW 316), the Adage computer (AGT/130), and the transponder data system (TDS).

The required inputs to the TAATM program consist of aircraft-state data and the echoing, for verification, of controller messages. The aircraft-state data are available from the HW 316 and the TDS. The HW 316 provides the following data:

1. X-position
2. Y-position
3. Z-altitude
4. Radar ground speed
5. Heading

These values range in word size from 11 bits to 18 bits. The TDS down-link provides the following data to the TAATM simulation program:

1. Indicated airspeed
2. X and Y (velocity components relative to Earth)
3. Bank angle

These values range in word size from 12 bits to 16 bits. Input from the Adage computer to the TAATM simulation program consists of a verification of TAATM-supplied controller messages. This "echoing" technique assures the TAATM simulation program that the controller messages were correctly received and processed by the Adage computer.

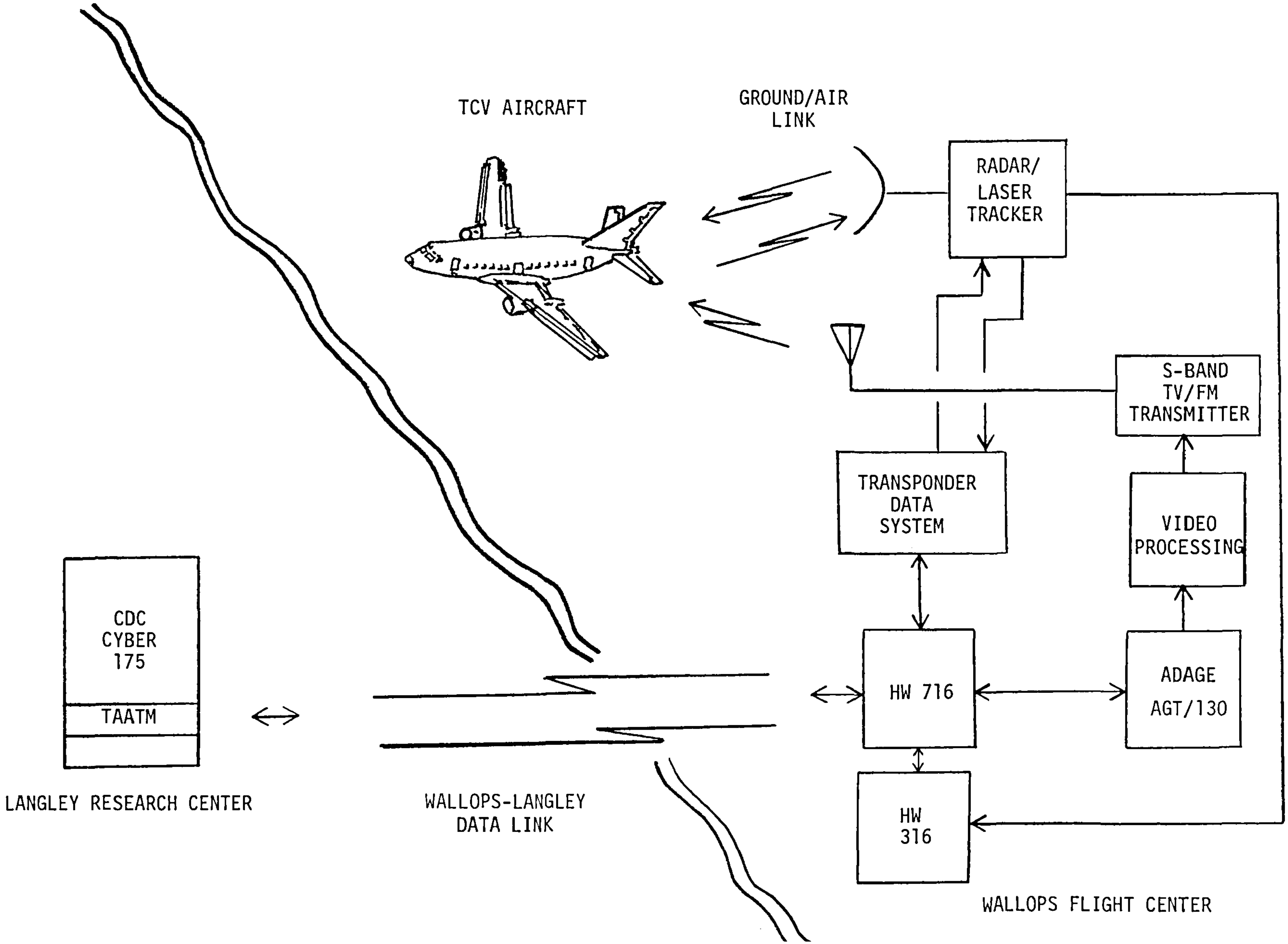


Figure 2.- Wallops-Langley tie-line system components and data paths.

The total of inputs from these three sources is transmitted by the HW 716 to the TAATM simulation program in one transmission block, consisting of 36 data characters. Each data character is comprised of six data bits, a parity bit, and a binary-indicator bit that allows for data transparency. Each input block is sized such that up to three retransmissions can occur if the initial transmission is erroneous.

Output from the TAATM simulation program consists of ATC information plus the position data for up to 40 simulated aircraft. The ATC information includes such data as speed change, direction change, and hold message. This information is provided for the simulated traffic as well as the TCV aircraft, and represents up to 10 different controllers. Additionally, information on the number of active aircraft in the simulation and the identification number of the controller responsible for the TCV aircraft is included. The data for the simulated aircraft include aircraft identification, velocity, heading, altitude, and bank angle.

The output data are sent to the HW 716 by TAATM in fixed-length blocks. The blocks contain 342 data characters, plus the protocol characters. Each data character consists of 6 bits of data, a binary indicator bit, and a parity bit. There can be up to 3 output blocks per scan, depending on the number of simulated aircraft. If there are 20 or less simulated aircraft, 2 output blocks are sent. If 21 to 40 aircraft are being simulated, 3 output blocks are sent. The output blocks are set up on a priority basis with the number of retransmissions based on the priority a block is assigned. Block 1, containing the ATC instructions/commands, is the highest priority block. Block 1 data can be transmitted up to 5 times, that is, 4 retransmissions if required. Block 2 data are transmitted next, with up to 3 retransmissions if required. Block 3 data, if required, are transmitted last with a possible 2 retransmissions (ref. 1).

The TAATM simulation program determines the number of times an input block can be retransmitted and establishes a priority method for the transmission of output blocks. TAATM's input- and output-block handling strategies are influenced by the following factors:

1. Expected telephone-line error rates

2. Data transmission rates and block size

3. Radar 4-second sweep time

The expected telephone-line error rate is approximately 1 in 10^5 bits. The data transmission rate on the telephone line is 4800 bits per second, which translates to 600 characters per second. This line speed, coupled with a maximum physical block size of 1040 data characters, as dictated by the communications protocol, is sufficient to meet the data volume requirements of the TAATM simulation program. A statistical analysis of transmission error effects on data transfers in a real-time environment is described in reference 2. Regarding the radar sweep time, the 4-second interval is partitioned by TAATM to allow for the input, computational, and output phases of the program. Based on the data requirements and the results published in reference 1, the TAATM program allocates approximately 406 milliseconds to successfully input (retransmit if necessary) the data from Wallops, and approximately 2.84 seconds to successfully output the computed results to the Wallops facility. Unsuccessful input or output attempts within a given 4-second interval allow the TAATM simulation program sufficient time to terminate processing activities for that interval and resynchronize for the next 4-second radar sweep.

The data transparency (binary) requirement, along with the timing considerations that involve doing tie-line data transfers in a real-time environment, establishes certain design criteria for the tie-line communications protocol as well as the CYBER communications software.

TIE-LINE COMMUNICATIONS PROTOCOL

The tie-line communications system uses conventional hardware communications elements. These include a dedicated, 4-wire telephone circuit, 4800-baud synchronous modems (modulator/demodulator), and a communications line multiplexer. At the onset of this project, one of the first elements to be defined was the communications protocol. Once defined at Langley, the protocol was documented and sent to Wallops as a specification to be implemented on the Honeywell 716 computer. The protocol chosen was a modified version of the Control Data 200 UT protocol. The standard version of this protocol supports half-duplex, ASCII (American Standard Code for Information Interchange) character-mode communications and is described in reference 3. The modified version is similar to the standard version, with the exception that binary data can be transmitted on the communications link. Other minor modifications are included in the detailed description of the Langley/Wallops communications protocol found in appendix B.

BINARY INPUT/OUTPUT

The standard protocol does not permit the transmission of binary data. However, the TAATM simulation program has data requirements on both input and output to process binary data. The problem involving the protocol is that of keeping the possible data-character bit patterns from conflicting with the protocol control characters. The protocol control characters cannot be changed easily because the communications hardware at Langley and Wallops are standard products and use certain of the control characters in the management of the communications line. For example, the generation of the message-parity-check character of the protocol is a function of the communications-line controllers. Upon examining the bit patterns of the ASCII character set, it was determined that protocol-control-character conflicts can be avoided if the 8-bit data characters are constructed so that bit position 6 is always set to "1." This procedure was adopted, and the communications software on both ends of the tie-line assumed the bit-setting function.

CYBER COMMUNICATIONS SOFTWARE

The configuration of a CYBER computer primarily consists of a high-speed central processing unit (CPU) coupled with a set of independent, less-powerful peripheral processor units (PPU) for input/output. The commmunications software in the CYBER computer consists of a CPU portion and a PPU portion. The primary function of the CPU portion is to provide an interface between the TAATM application and the PPU software responsible for the physical bidirectional flow of data on the tie-line. It was in the PPU portion that certain innovative techniques were implemented that significantly improved the debugging and quality-assurance cycles and enhanced future expansion potential.

PPU COMMUNICATIONS DRIVER

The program that controls the actual physical transfer of data on the tie-line operates in a peripheral processor unit (PPU). Within the CYBER 175 architecture, PPU programs are primarily used for input/output activities and system-control functions. PPU programs, often referred to as "drivers," operate independently of the CYBER 175 CPU. This independent operation, however, is often in support of CPU program activities, as is the case with the tie-line PPU driver (named UWD) and the TAATM simulation program. Since the primary task of the UWD program is the time-critical transfer of data between the TAATM program and the HW 716 at Wallops, UWD program design minimizes the data-handling and data-formatting tasks by assigning these tasks to the CPU software. The offloading of these tasks not only enhances UWD execution time, but provides the TAATM program with protocol decision-making and control capabilities.

The concept of control is exemplified by the fact that UWD is table-driven by the CPU. This table, referred to as the I/O (input/output) table, consists of 2N + 2 entries; N refers to the number of communication lines that the program UWD is expected to support at any one time. The maximum number of lines is 8. Each entry contains 60 bits of information, although the entire 60-bit entry is not always used. The first word of the table contains the UWD program control word. It is set by a CPU communications routine to initialize and terminate the UWD program. It is also used by the UWD program to acknowledge functions set for it by the CPU communications software. The second entry in the table contains values that indicate errors, primarily those associated with the multiplexer. These errors are evaluated for possible recovery action by the CPU portion of the communications software. Most of these errors are fatal and require corrective maintenance. The remaining values in the table are used as word-pairs. Each word-pair supports an active communications line. The first entry of the word-pair contains six parameters used for information associated with each individual line. These parameters include the size and address of the central-memory I/O buffer, the number of characters transferred during an input or output process, the port number on the multiplexer, and the status on the normality or abnormality of a given I/O process. The second entry of the word-pair contains the timing information that the program UWD uses to time-out on line errors, I/O operations, and character transfers. The times are in milliseconds and are set by the CPU portion of the communications software. Appendix C describes the information and format of the I/O table.

The TAATM program, using the I/O table via flag words, directs the I/O activity of program UWD. Knowing its time position within the 4-second radar sweep interval, TAATM sets and clears the input and output flags, and controls the output-block selection. Since TAATM controls its real-time environment, the operation of program UWD is started and terminated by TAATM as required. This means that the UWD program uses a PPU only when required by TAATM for tie-line input/output. At other periods of time during which TAATM is operational (i.e., for edit mode changes), the PPU is available for other CYBER PPU activities.

Conventional methods for debugging PPU programs, especially for examining the data content of a PPU buffer, can be time-consuming. Generally, to dump a CYBER PPU program requires a system "deadstart," a process normally used to make the CYBER operational for job processing. This process was seldom needed during the development and checkout of the UWD program. All the tie-line data, including the communications protocol, were visible at the CYBER console, since central-memory buffers were used. This visibility reduced the checkout time of the tie-line communications software at Langley and Wallops. The communications equipment at Wallops did not permit visual observation of the tie-line transactions. Consequently, during tie-line checkout, Langley, using the CYBER console and a telephone voice line to Wallops, relayed to Wallops the various protocol and data messages that the CYBER transmitted and received. Additionally, Langley intentionally caused error conditions by entering invalid data or protocol messages via the CYBER console, and thereby expedited the checkout of error-recovery procedures.

CPU COMMUNICATIONS SOFTWARE

The CPU portion of the communications support software consists of a number of subroutines. The major purposes of the CPU routines are

1. To interface the TAATM application to the PPU program UWD in a real-time environment

2. To provide general support relating to tie-line activities

Regarding the first item, there are three routines that function as an interface to TAATM. The first routine, Z3STRT, was developed to permit TAATM, prior to real-time operation, to initialize tie-line transmissions or to reestablish tie-line communications if the communications are interrupted by an error or malfunction in the communications system. This software sets the proper table values for TAATM and causes the UWD program to transmit the clear-write protocol message which initializes tie-line dialogue. The second routine, Z3SEND, processes the data-transmission request by TAATM. This routine, driven by the input and output flags set by TAATM, is segmented to operate in real time. TAATM calls this routine each real-time frame, 128 calls per 4-second radar sweep. The segmentation of this routine guarantees that TAATM maintains control throughout the 4-second interval. The functions of Z3SEND include managing the TAATM data buffers, tracking the status from UWD on each transmission block, and interfacing to the data-preparation routines that format the input and output tie-line data. The third interface routine, AGTO, is called by the real-time supervisor whenever TAATM begins operating in real time. The function of routine AGTO is to examine the I/O table to see if program UWD needs to be operating with TAATM. If UWD is required and not already operating in a PPU, then AGTO starts UWD in a PPU. This situation could occur when, subsequent to using the tie-line in real time, TAATM executes in a non-real-time mode, that is, to edit-in a program change. This execution mode change causes the UWD program to exit the PPU and to be restarted by AGTO upon the return of TAATM real-time operations.

The general support routines were developed principally to support TAATM in the categories of input, output, protocol management, and error recovery. These routines typically do not interface to TAATM directly, but operate under the control of, or in conjunction with, the interface software routines previously discussed.

Subroutines OLIP and OLOP are the prime components of the input/output support categories. Subroutine OLIP processes the input data from Wallops by transferring the tie-line data received by the PPU program UWD into a buffer suitable for processing by the TAATM program. Upon calling routine OLIP, TAATM provides OLIP with the number of 16-bit words expected from the HW 716 and receives upon return from OLIP a central-memory buffer in which each word contains eight 6-bit binary characters. On output from Langley to Wallops, subroutine OLOP is used by TAATM to format the data for tie-line transmission. TAATM provides the OLOP routine with an output buffer formatted into eight 6-bit characters per central-memory word. The OLOP routine prepares these data by transforming each 6-bit character into an 8-bit character suitable for transmission. This transforming process includes the setting of the binary bit and the character-parity bit. This sequence of 8-bit characters is further enveloped with the proper protocol characters and placed into a central-memory buffer for transmission by UWD. Figure 3 illustrates the formatting technique used for the tie-line data. In the figure, the solid line represents output data flow, Langley to Wallops. The dotted line represents input data flow, Wallops to Langley.

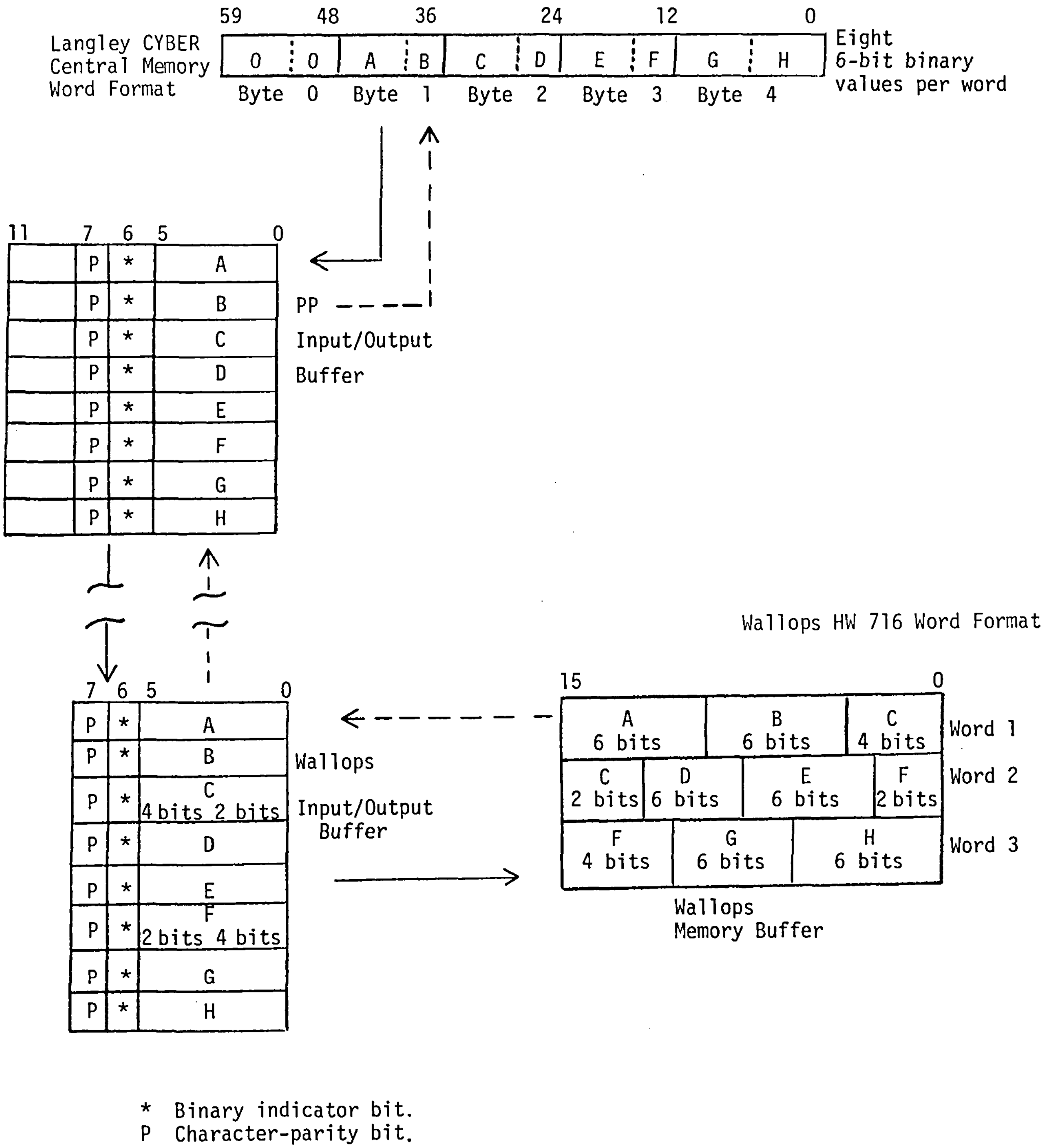


Figure 3.- Wallops-Langley buffer formats.

The function of managing the protocol is provided by subroutine OLPH. Its other functions include keeping track of the station address (toggling it when required), handling the retransmission requests, placing the proper buffer address in the I/O table (for UWD), and checking the I/O table for the various error statuses reported by UWD. If an error is found, an error-analysis routine is called. This routine, named OLERROR, reports errors concerning the multiplexer, the tie-line telephone circuit, or the protocol. When an error occurs, OLERROR identifies the problem, selects the proper error message, and returns control to the error-processing routine (RTERROR) of the real-time simulation subsystem. An orderly recovery process is initiated. If the error type is nonfatal, the TAATM program attempts immediate recovery. Nonfatal errors are often related to timing restraints placed on the tie-line, that is, a lack of a communications protocol response within a given time frame. Fatal errors, generally related to communications hardware malfunctions, require further diagnostic analyses. If errors do occur during a simulation run, their number and type are recorded for post-run analysis. Results of the analysis determine whether certain hardware diagnostics should be used and whether software adjustments need to be made.

CONCLUDING REMARKS

The goal of designing and implementing a telecommunications interface that is operable in a real-time environment has been accomplished. The first successful flight test was made in July 1979. The test incorporated the major components of the experiment: the Terminal Configured Vehicle (TCV), the Wallops Flight Center, the Wallops/Langley tie-line, and the Terminal Area Air Traffic Model (TAATM) simulation program. There were two additional flight tests later in 1979. The tie-line development effort resulted in a peripheral processor unit (PPU) program design that employs a concept of protocol independence. Since the protocol is managed in the central processing unit (CPU), including the control characters, future protocols could be accommodated with possible adjustments made only to the PPU/line-multiplexer interface.

Langley Research Center
National Aeronautics and Space Administration
Hampton, VA 23665
November 25, 1981

APPENDIX A

REAL-TIME SIMULATION SUBSYSTEM

The Langley Central Scientific Computer Complex contains eight Control Data Corporation computer systems, linked together via a common permanent file system (fig. A1). This facility supports a myriad of aerospace research applications and requires a variety of peripheral equipment and subsystems. A major subsystem, and one that has dictated many configuration and performance requirements of the computer complex, is the real-time subsystem. Figure A2 shows the hardware configuration of one simulation subsystem. The computer complex presently has two simulation subsystems, each capable of concurrently supporting three simulation applications. The two CYBER 175 computers support the real-time subsystems. Figure A3 depicts a CYBER 175 real-time machine environment. Each real-time simulation program operates in one of the CYBER 175 machines. Specifically, each program operates within a logical area called a control point. Each simulation application specifies a time in which the input-compute-output cycle must be performed. This time is referred to as the "frame" time and can be set to binary intervals between approximately 1 millisecond and 1 second. The Terminal Area Air Traffic Model (TAATM) simulation program uses a frame time of 31.25 milliseconds. Operating in conjunction with the simulation-application program are simulation-control programs required for real-time operation. Two of these programs reside in peripheral processor units (PPU). One of these programs provides the input data from the analog-to-digital equipment to the simulation-application programs. The program operates under a timing sequence developed by the simulation-scheduling program. Another peripheral processor program under timing control performs the necessary output functions between the simulation applications and the digital-to-analog equipment. There is, in addition to these peripheral processor input/output programs, a simulation-scheduling program and a supervisor program. The simulation-scheduling program is a central-memory program that schedules the simulation applications to operate in a real-time environment under control of the CYBER system monitor and the aforementioned peripheral processor unit (PPU) input/output programs. The scheduler develops a master timing cycle which insures against timing conflicts and guarantees the successful operation of the real-time events. The supervisor program, an integral part of each simulation-application program, provides system-linkage and data-formatting functions. This program eliminates any special programming by the simulation-application programmer. One additional component that is essential to the real-time simulation subsystem, but not actually part of the simulation subsystem, is the interactive-job processor. Interactive terminals are used to start and terminate simulation applications as well as to modify, if necessary, the simlulation-applications program. In summary, the simulation-control programs insure the real-time integrity for the overall simulation subsystem and link each simulation application to the simulation equipment. A detailed description of Langley's real-time simulation facilities is presented in reference 4.

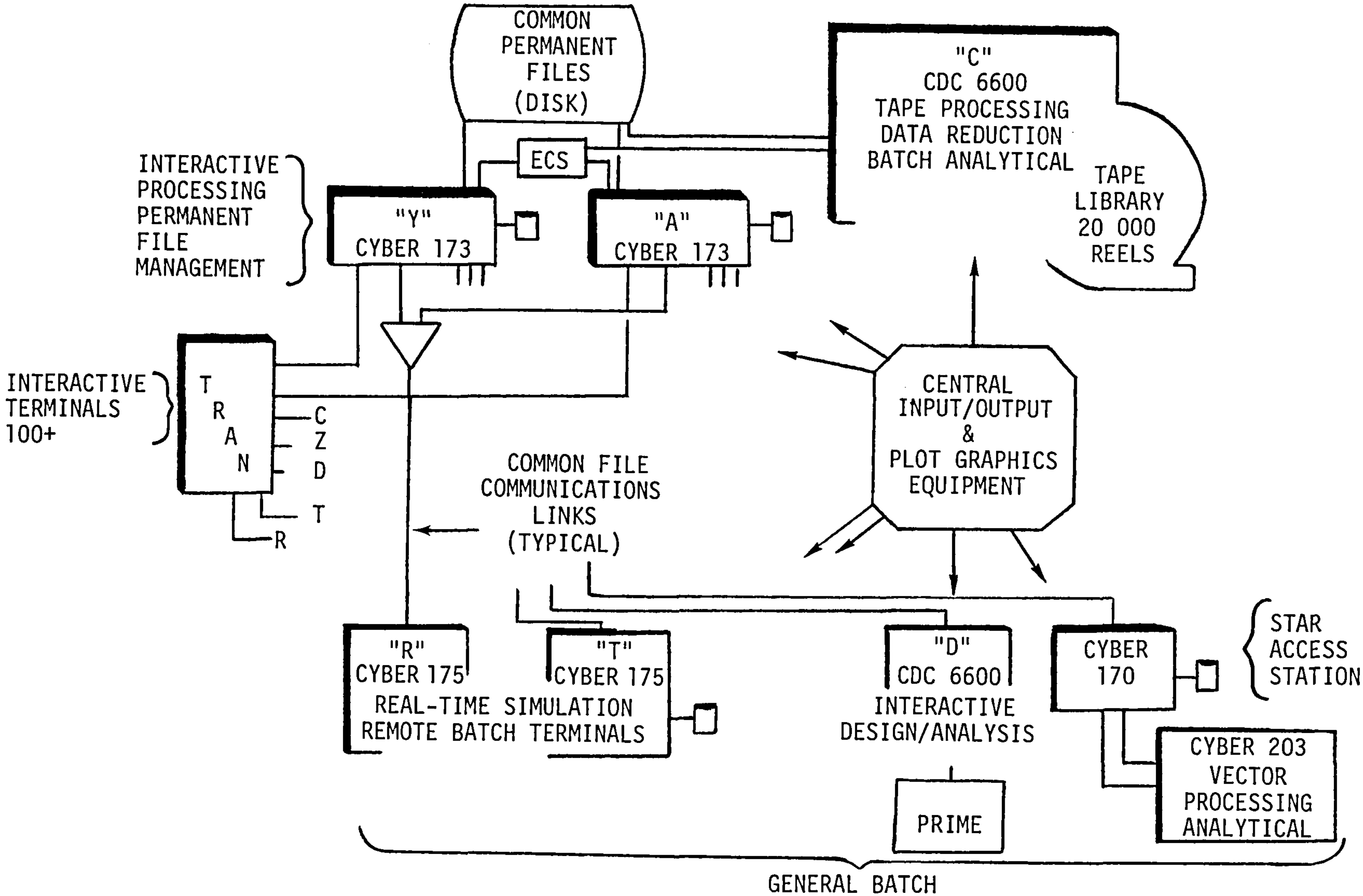


Figure A1.- The Langley Central Scientific Computer Complex.

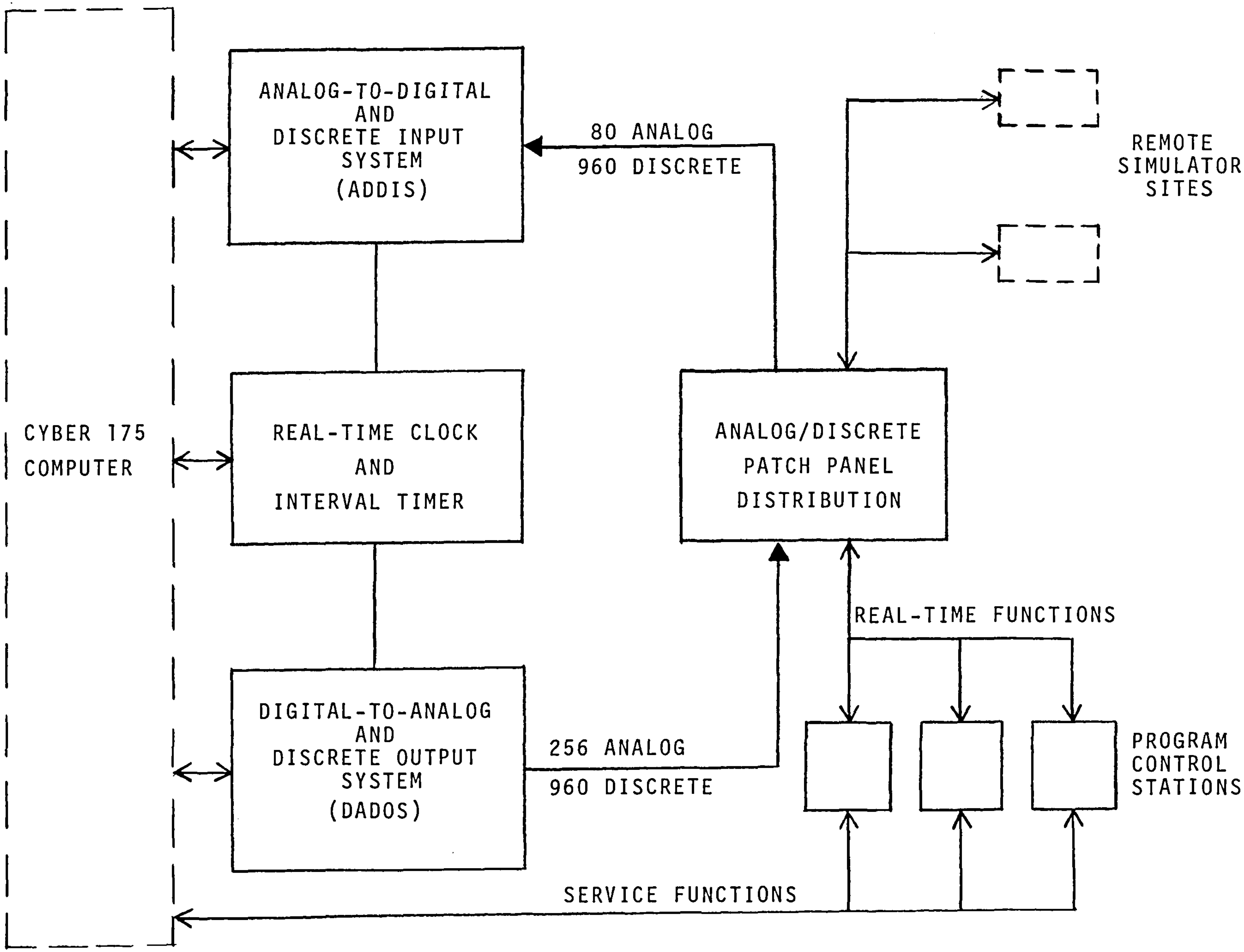


Figure A2.- Simulation subsystem.

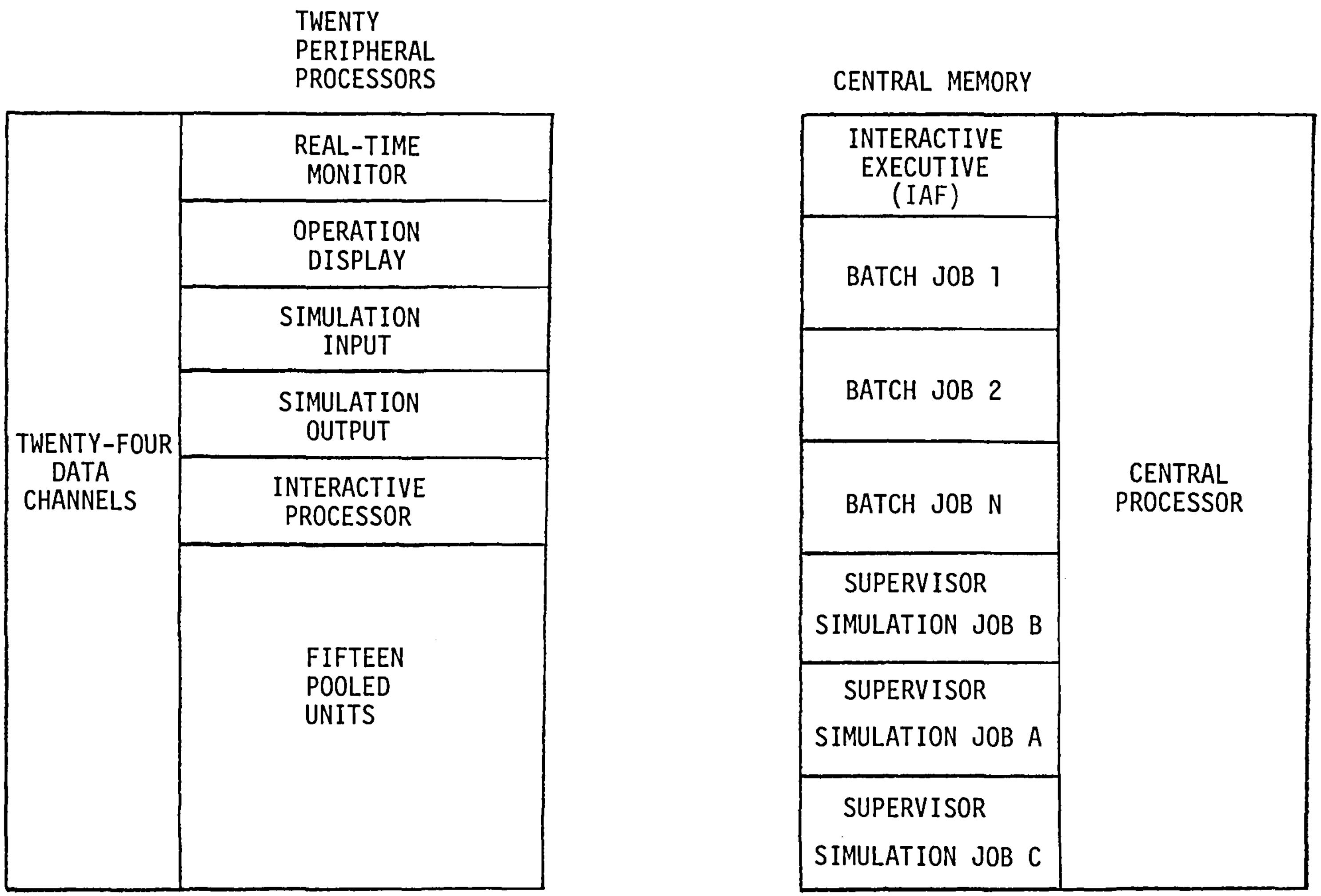


Figure A3.- CYBER 175 real-time simulation machine environment.

APPENDIX B

DESCRIPTION OF DATA-LINK COMMUNICATION PROTOCOL

BETWEEN LANGLEY AND WALLOPS

This appendix outlines functional descriptions, byte format, message format, types of messages, code repertoire, and message processing of the Control Data 200 User Terminal protocol (CDC 200 UT) as adapted to the Wallops Flight Center/Langley Research Center communications link.

Data-Link Control Procedures

General function.- The CDC 200 UT protocol is used for communication between the Wallops site and the host. The host is defined as the Langley Central Scientific Computer Complex. The major functions accomplished by the protocol system (hardware and software) are as follows:

1. Interpret messages received
2. Transmit messages
   a. The host is dominant (initiates activity)
   b. Wallops Flight Center is subordinate
3. Check and generate character and message parity
4. Check for correct control characters in messages received
5. Append correct control characters to messages transmitted
6. Detect errors in messages received
   a. Parity
   b. Unrecognizable control codes
   c. Improper termination of messages
7. Respond to accurate reception of data by an acknowledging transmission
8. Respond to an inaccurate reception of data by an error transmission or by a time-out (no response)
9. Retransmit in accordance with error-recovery procedures

Byte Format.- Communications between the central computer system and Wallops are bit serial, byte serial. Each byte is 8 bits including one parity bit. Bit position 7 is defined as the parity bit. Bit position 0 is defined as the least significant bit. A bit is transmitted and received as either a 1 or 0. Bit position 0 is received and transmitted first. The order of 1's in bit positions 0 to 6 determines the code represented by the byte. Combinations of these bits enable a total of 128 different codes, but only 64 different codes may reside within the binary data format. The transmitting routine counts the number of 1's in bit positions 0 to 6 of a generated byte to determine the state of the parity bit. If the number is even, it sets a 1 for the parity bit. If the number is odd, it sets a 0 for the parity bit.

In both cases, the number of 1 bits in the byte will be odd. Upon receipt, the receiving routine counts the number of 1's in all bit positions, including the parity bit. If the number of 1's is even, it initiates an error sequence. Where applicable, the ASCII (American Standard Code for Information Interchange) character set is used. The octal representation of the ASCII protocol characters is provided later in this appendix.

General message format.- The following elements make up a CDC 200 UT transmission:

1. Sync codes

2. Opening control characters

3. Data characters (optional)

4. Closing control characters

5. Message parity character

The general message format is shown as the following sequence of characters:

```
      Sync
      Sync
      Sync
      Sync
Start of header    (SOH)
Site address       (SA)
Station address    (STAT ADR)
Control code       (CC)
•  Data
•      and
•         escape
•               characters
ASCII end of text (ETX)
Message parity character (MPC)
```

All messages transmitted are preceded by a minimum of four sync codes to assure synchronization recovery on the receiving end. Sync codes may appear anywhere in the message except between the start-of-header code (SOH) and the control code (CC) or between the end-of-text character (ETX) and the message-parity character (MPC). The start-of-header code informs the receiving device that the following two codes are addresses. Site and station addresses follow in that order. The site address designates an on-line site element to which a message is addressed or from which a message is received. Since the Wallops site element is a single-station device, the station address is defined mainly for purposes of program compatibility with the multistation devices. The address is normally 141_8 or 161_8 and has special applications when used with write messages and responses. The control code defines the intent of the message:

POLL, WRITE, CLEAR WRITE, DIAGNOSTIC WRITE, REJECT, READ, ACKNOWLEDGE, and ERROR

These messages are described in this appendix and are herein after represented as all capitals to denote a message type.

If data are included in a transmission, the data characters follow the control code. The number of data characters may range from 0 to 1040_{10}. Following the data-character block is the escape code and an end-of-data indicator. Data compression/decompression techniques are not used.

Messages are terminated by an ASCII end-of-text character and then by the message-parity character. Message parity is applicable from the start-of-header code through the ASCII end-of-text code, excluding all sync codes. It insures that an odd number of 1's have been transmitted in each bit position of all bytes. The parity bit of the message-parity character is parity for that byte only.

The 7-bit octal representation (without parity) of the ASCII protocol characters is as follows:

Sync	026	POLL	005
Start of header	001	ACKNOWLEDGE	006
READ	023	REJECT	030
WRITE	021	ERROR	025
DIAGNOSTIC WRITE	024	Escape	033
CLEAR WRITE	022	End of text	003

Message flow.- The host initiates all transmission activity. The Wallops site responds to host communications.

The host sends only the following messages:

1. POLL

2. WRITE

3. DIAGNOSTIC WRITE (used by diagnostic programs only)

4. CLEAR WRITE (used for sign-on and recovery)

The Wallops site responds with one of the following messages:

1. REJECT

2. READ

3. ACKNOWLEDGE

4. ERROR

The basic function of each message is as follows:

POLL

The host sends a POLL message to the Wallops site periodically for the following reasons:

1. The POLL asks if the Wallops site has any data to send. If the Wallops site has data ready, they are sent to the host in a READ message, which is acknowledged by the host with a WRITE. This WRITE may be a zero-length WRITE if the host has no data to send, or it may contain valid data. If there are no data to send from Wallops, the response is a REJECT message.

2. The POLL is a means of maintaining communication between the host and the Wallops site. Each knows the other is still functioning because of the periodic transfer of at least POLL and REJECT. Periodic polling is an optional feature.

REJECT

The Wallops site sends a REJECT message to the host when one of the following conditions exists:

1. A host POLL has been received, but the Wallops site has no data ready to send

2. A host WRITE message has been received, but the Wallops site could not accept the data

READ

After receiving a POLL from the host, the Wallops site may send data to the host via a READ message.

ERROR

The following error conditions in messages received by the Wallops site cause an ERROR message to be sent to the host:

1. Parity-error detection by the Wallops site on a character received after the site-address code

2. Message-parity character incorrect

3. Illegal control code

4. Illegal station address

5. Improper sequencing of control characters received within a transmission block (after site-address code)

WRITE

The host uses a WRITE message for two purposes:

1. To send data to the Wallops site

2. To acknowledge accurate reception of a READ message from the Wallops site

The host answers an accurately received READ message with a WRITE message. This indicates to the Wallops site that the data in the READ message were accurately received by the host. The WRITE message may or may not contain data. A zero-length WRITE is sent by the host if there are no data to send.

DIAGNOSTIC WRITE

After receiving a DIAGNOSTIC WRITE from the host, the Wallops site must send a READ message to the host. The READ message contains the data from the DIAGNOSTIC WRITE. If a parity error is detected in a data character received, the character in error is replaced by an error code in the read message sent to the host. The error code is equivalent to the ASCII character SPACE (40_8). If the parity error is detected in a character outside the data field, the ERROR response message should be sent to the host. The DIAGNOSTIC WRITE is an optional feature.

CLEAR WRITE

In order to implement a sign-on capability, the CLEAR WRITE message will be sent by the host computer to the Wallops site. Receipt of this message would mean that Langley would like to initialize input/output (I/O) operations with the Wallops site. The Wallops site replies to the CLEAR WRITE in the same manner that it would reply to a WRITE message (ACKNOWLEDGE, REJECT, ERROR). For the sake of convention, the Wallops site will not be considered signed-on until an ACKNOWLEDGE is returned. No data will be sent with the CLEAR WRITE message. The CLEAR WRITE is also sent by the host computer to reinitialize after a time-critical malfunction has occurred.

ACKNOWLEDGE

The Wallops site sends an ACKNOWLEDGE message to the host when a WRITE message has been accurately received.

Protocol Implementation Features

No response.- If the Wallops site detects a transmission error before the station-address character, no response is sent to the host. The host will then time-out and repeat the transmission (POLL or WRITE).

Station-address sequencing.- The station-address character is effectively a transmission sequence number. Its purpose is to validate the flow of transmission blocks between the host and the Wallops site. The station address has been defined as 140_8, 160_8, 141_8, and 161_8.

The station address toggles between a $14X_8$ and a $16X_8$ (X = 0 or 1) as follows:

1. The initial state of the station address is $14X_8$
2. The state of the station address changes to $16X_8$ when the host receives an ACKNOWLEDGE transmission (in response to a WRITE)
3. Thereafter, each time the host receives an ACKNOWLEDGE, the state of the station address is toggled

When the Wallops site responds to a WRITE with an ACKNOWLEDGE, the state of the station address sent is the same as was received. When the Wallops site must respond to a WRITE with a REJECT or an ERROR, the state of the station address sent in the response must be opposite the state of that received. The 0 (even) state is used in the POLL message. It is also used in the REJECT response to a POLL and the ERROR response to a POLL. The 1 (odd) state is used in a REJECT to a WRITE or an ERROR response to a WRITE message. The 1 (odd) state is also used in the READ, WRITE, and ACKNOWLEDGE messages.

Two consecutive WRITE messages with the same station address indicate that the host either received an ERROR or REJECT reply or could not interpret the response; therefore, the data in the current block are a retransmission of the data in the previous block. The various states of the station address are summarized in the following table:

Host sends		Wallops replies			
Message	Station address	Read	Reject	Error	Acknowledge
POLL	140	141	140	160	(a)
WRITE	141	(a)	161	161	141
POLL	160	161	160	140	(a)
WRITE	161	(a)	141	141	161

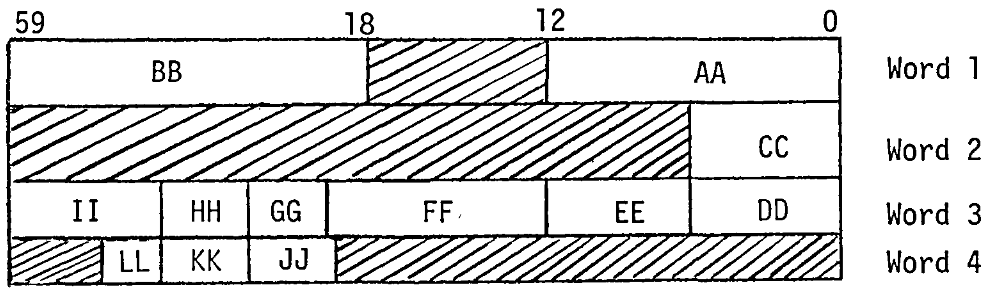
[a]Invalid response.

<u>Response times</u>.- The Wallops response times to protocol messages generated by the host are flexible but are expected to be between 50 and 150 milliseconds.

APPENDIX C

CYBER COMMUNICATIONS SOFTWARE I/O TABLE

The following information describes the input/output (I/O) table used by the CYBER tie-line communications software. The I/O table as configured assumes only one active line, words 3 and 4 would be repeated for additional active lines. Cross-hatched areas in the I/O table represent unused bit positions.



Word 1: AA - Program UWD control word
- 0 = Initiate program UWD
- 2 = Terminate I/O activity and exit

BB - logical name of multiplexer

Word 2: CC - UWD Program error code
- 1 = Invalid number of lines
- 2 = Invalid call to program UWD
- 3 = No multiplexer file name table found
- 4 = Cannot connect to multiplexer
- 5 = Program UWD already running
- 6 = Equipment status table error
- 21 = Input-function failure
- 22 = Output-function failure
- 24 = Multiplexer failure during input
- 25 = Multiplexer failure during output
- 26 = Multiplexer failure during status input
- 27 = Multiplexer memory parity error
- 30 = Multiplexer output service error

Word 3: DD - Bits 0 to 5 (I/O status)
0 = I/O initialized on this port
1 = I/O completed abnormally
3 = I/O completed normally

EE - Bits 6 to 11 (Port error code)
Error occurred on this particular port
1 = Lost carrier error
2 = Data lost - serviced too slow
3 = Data lost due to storage move
4 = Buffer field length error
5 = Character time-out
6 = Phase time-out
7 = Transmission parity error
10 = Port assignment error
11 = No end of text

FF - Bits 12 to 23 (Number of characters transferred in an I/O phase)

GG - Bits 24 to 35 (Size in central-memory words of linear I/O buffer)

HH - Bits 36 to 47 (First word address of I/O buffer)

II - Bits 48 to 59 (Port number)

Word 4: JJ - Bits 23 to 35 (Maximum number of milliseconds before time-out on any line errors)

KK - Bits 36 to 47 (Maximum number of milliseconds before time-out on input/output operations)

LL - Bits 48 to 49 (Maximum number of milliseconds the peripheral processor unit (PPU) will wait between characters in the input or output phase)

REFERENCES

1. Montoya, R. J.; and Stroh, R. W.: Research on Airborne Traffic Situation Displays. Volume II - Integration of the Terminal Configured Vehicle (TCV) With the Wallops Flight Center Experimental Research Facility (WFCERF). NASA CR-159256-1, 1980, pp. 128-130.

2. Credeur, Leonard: Analysis of Transmission Error Effects on the Transfer of Real-Time Simulation Data. NASA TM X-74019, 1977.

3. Control Data® 734 Batch Terminal - Reference Manual. Publ. No. 62971300, Control Data Corp., c.1974.

4. Eckhardt, Dave E., Jr.: Description of Langley Research Center Computer Complex and Special Features for Real-Time Simulation Applications. NASA paper presented at the Eastern Simulation Council Meeting (Hampton, Virginia), Sept. 26, 1968.

1. Report No. NASA TM-83231	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle DESIGN AND IMPLEMENTATION OF A TELECOMMUNICATION INTERFACE FOR THE TAATM/TCV REAL-TIME EXPERIMENT		5. Report Date December 1981 6. Performing Organization Code 992-16-05-03
7. Author(s) Joseph D. Nolan		8. Performing Organization Report No. L-15021
9. Performing Organization Name and Address NASA Langley Research Center Hampton, VA 23665		10. Work Unit No. 11. Contract or Grant No.
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546		13. Type of Report and Period Covered Technical Memorandum 14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

The traffic situation display experiment of the Terminal Configured Vehicle (TCV) research program requires a bidirectional data communications tie-line between the Wallops Flight Center Experimental Runway Facility and the Langley Central Scientific Computer Complex. The tie-line is used in a real-time environment on the CYBER 175 computer by the Terminal Area Air Traffic Model (TAATM) simulation program. Aircraft position data from Wallops are processed by TAATM with the resultant output sent to the Wallops facility for the generation of air-traffic situation displays, which are transmitted to a research aircraft. This report describes the design and implementation of the Langley communications software system that supports the traffic situation display experiment in a real-time environment.

17. Key Words (Suggested by Author(s)) Data communications Real-time air-traffic displays	18. Distribution Statement Unclassified - Unlimited Subject Category 04

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	24	A02

For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1981

National Aeronautics and
Space Administration

Washington, D.C.
20546

Official Business

Penalty for Private Use, $300

THIRD-CLASS BULK RATE

Postage and Fees Paid
National Aeronautics and
Space Administration
NASA-451

U.S.MAIL

NASA

POSTMASTER: If Undeliverable (Section 158
Postal Manual) Do Not Return