AIRCRAFT INTERROGATION AND DISPLAY SYSTEM:
A GROUND SUPPORT EQUIPMENT FOR DIGITAL FLIGHT SYSTEMS

Richard D. Glover

April 1982
AIRCRAFT INTERROGATION AND DISPLAY SYSTEM:
A GROUND SUPPORT EQUIPMENT FOR DIGITAL FLIGHT SYSTEMS

Richard D. Glover
Ames Research Center
Dryden Flight Research Facility
Edwards, California
INTRODUCTION

The National Aeronautics and Space Administration (NASA) is conducting research in many areas involving advanced digital systems for both manned and unmanned aircraft, and in ground-based simulators. As these various types of digital flight systems have become more complex, the need has arisen for more sophisticated ground support equipment (GSE) for systems integration, software verification and validation, pre- and postflight testing, and system maintenance. Until recently, the approach taken was for each project to procure special purpose GSE, resulting in a multiplicity of different types of equipment of varying capability. These types of GSE generally were single purpose and were surplussed at the termination of the project. Usually, none of the GSE development investment could be recouped for the next project.

As an approach to a resolution of this problem, the NASA Dryden Flight Research Facility undertook the development of a microprocessor-based user-programmable general purpose GSE, termed aircraft interrogation and display system (AIDS). A prototype was constructed, interfaced with the F-8 digital fly-by-wire (F-8 DFBW) iron bird simulator, and used successfully to support F-8 flight software verification and validation. The general purpose utility of the AIDS was confirmed when applied to the highly maneuverable aircraft technology (HiMAT) project. Using new software, the prototype was easily interfaced with the HiMAT aircraft, and it quickly demonstrated its capability by providing a fortyfold increase in random access memory (RAM) data display bandwidth.

The utility of the AIDS during HiMAT flight control computer testing and systems integration validated the flexibility of the system and led to plans to apply it to other projects. Two AIDS systems are in service, and a third is under construction. The total number of present and planned users is five. This paper describes the AIDS design and mechanization, summarizes operational experience to date, and discusses plans for the future.

The use of trade names or names of manufacturers in this report does not constitute an official endorsement of such products or manufacturers, either expressed or implied, by the National Aeronautics and Space Administration.

SYMBOLS AND ABBREVIATIONS

A/D analog-to-digital converter
AFTI advanced fighter technology integration
AIDS aircraft interrogation and display system
ARW advanced research wing
ASCII American standard code for information interchange
ASEG absolute segment
C&D controls and displays
CPU central processing unit
CRT cathode ray tube (display)
CSEG control segment
DAC digital-to-analog converter
DAST drones for aerodynamic and structural test
DFBW digital fly-by-wire
DSEG data segment
DSPM dispersed sensor processing mesh
GSE ground support equipment
HiMAT highly maneuverable aircraft technology
HSMU high speed math unit
I/F interface
I/O input/output
KB keyboard
LED light-emitting diode
MDS microprocessor development system
PROM programmable read-only memory
RAM random access memory
RTMTX real-time multitasking executive
STC system test console
TTL transistor-transistor logic
USART universal synchronous/asynchronous receiver-transmitter

EXPERIENCE WITH SPECIAL PURPOSE GSE

A significant amount of experience was gained during the F-8 DFBW program in the formulation and use of display and driver GSE devices for flight control design, development, verification and validation, troubleshooting, maintenance, preflight testing, and research experimentation (ref. 1). The ground display software was implemented in the F-8 DFBW flight computer itself and consisted of several dedicated and special purpose displays, including system redundancy management status, dynamic sensor data, aircraft system status, failure status, and preflight test and maintenance results.
Although the display system was highly refined and was a key element in the successful development of the fly-by-wire system, it had several drawbacks. First, the display system was designed to operate integrally with the triple-redundant digital fly-by-wire control system, and as such it had to be nonintrusive; that is, the display functions could not alter flight control system operation. This complicated the display system software. Second, the display system required a modest but not negligible share of the flight computer cycle time and memory resources. Third, the display software required a relatively high level of verification because it resided in the flight computer, even though it was never executed in flight. Finally, the system was not portable, and it could not be used on other aircraft programs.

The driver software used for verification and validation tests, such as triplex sensor fault detection, isolation, and recovery, was implemented in the mainframe computer used for aerodynamic simulation. Special purpose pulses, waveforms, and noise signature signals were generated by the driver software and interfaced to the flight computer sensor input processor. Although highly successful, this approach required substantial amounts of simulation computer time for relatively simple computational tasks at a time when the simulation computer served multiple users.

The experience, advantages, and disadvantages of the various approaches used on the F-8 DFBW program, as well as other flight system research projects, laid the foundation for the AIDS design.

DESIGN OBJECTIVES

The AIDS was originally conceived as a stand-alone general purpose ground support equipment device for aircraft digital flight control systems that had the display and driver capabilities of the GSE used for the F-8 DFBW. Early in the conceptual design it was determined that many other applications would be possible for this device. For that reason, design objectives were established that would guarantee the system's generality and flexibility. These design objectives included:

Mobility. The system should be capable of being moved between laboratories, iron bird, and aircraft.

Flexible input/output. The system should be easy to interface with digital and analog systems, be independent of the system-under-test architecture, and minimize that system's servicing burden.

Common core software support package. The system should provide a large share of commonly used display and driver functions for digital flight systems, including (a) number conversion to any desired format and engineering units calibration, (b) bit unpacking and display as event, (c) snapshot block data, (d) parameter trace, (e) data recording or plotting as stripchart or X-Y parameters, and (f) waveform drivers for redundant flight control sensors.

User-oriented displays. The displays should have dynamically refreshed display and provide for user formatting and labeling. Free-form display formats should be available that can be easily constructed in real time (during a test procedure) as new requirements develop. The operator should have the ability to interrupt a display at any time, make modifications to the format, and resume the display within a few seconds. In addition, the operator should have the ability to make display hard copies at any time. Such hard copies should be labeled with date, time, test title, and any other user-determined information.
Utilization of commercial components. Where possible, the system should use commercially available card-level microcomputer hardware and commercial software. This enhances long-term maintenance and minimizes development costs.

Speed. The system should be able to service flight control systems with cycle rates on the order of 50 to 100 samples per second.

Synchronization. The system should acquire and display snapshots of several data words occurring within one computer cycle frame (10 to 20 milliseconds).

Maintenance. The system should contain an integral diagnostic and maintenance support capability.

Operational modes. The system should be easily and quickly convertible between the operating modes shown in figure 1, including real-time data display, open-loop function generator, redundant sensor simulator, and simple closed-loop simulation (a simulation at a single flight condition with linear equations of motion).

FUNCTIONAL DESCRIPTION

The first AIDS device that was developed generally met the design objectives. The AIDS was designed around an 8085A microprocessor system. A diskette subsystem was incorporated which was fully compatible with the off-line support software used to create the AIDS software load modules. A commercially available real-time multitasking executive (RTMTX) was also incorporated, mainly for the management of the diskette drives and diskette directory services.

Figure 2 illustrates the functional arrangement of the AIDS. The particular operating mode is defined by the software modules contained on the system diskette. Any user displays that were previously created are stored on the scratch diskette. These two diskettes are accessed via the real-time multitasking executive software that is permanently recorded on programmable read-only memory (PROM) integrated circuits. The remaining system software is loaded from the system diskette by the RTMTX, and the display formats are loaded from the scratch diskette by the RTMTX as needed. The RTMTX then transfers control of the system to the software loaded, but remains available for subsequent diskette operations and other multitasking as required.

The AIDS supervisor module and companion operator input/output (I/O) modules are software that is common to all users. The supervisor provides command interpreting, software linking, a date register, an updated time-of-day register, and various system control functions. The I/O package provides the main operator interfaces to the control keyboard, the cathode ray tube (CRT) data display, and the hard copy peripherals. The operator enters system commands and display setup instructions via the control keyboard (KB). All displays are presented on the CRT display, which is refreshed at high speed on those areas of the screen which contain active (nonstatic) fields. Hard copies of any display may be made either by operator command or under supervisor control as desired.

User-unique software includes the user application supervisor, user timing control, and one or more user I/O modules. The user application supervisor provides servicing for user interrupts and interfaces with the RTMTX as required. The user timing control module provides basic timing for all user I/O and supporting computation. The user I/O servicing module services the data path to and from the system under test and provides for auxiliary analog outputs to nonAIDS peripheral devices as required.
Real-time data display:

Open-loop function generator:

Redundant sensor simulator:

Closed-loop simulation:

Figure 1. Examples of conceptual AIDS applications.
Figure 2. AIDS functional overview.
Figure 3 shows the mechanization of the current AIDS design. The entire system is mounted in a two-bay console that is mounted on wheels for mobility. The five major components are the computing subsystem, the I/O panel, the diskette drive subsystem, the operator terminal, and the line printer. The user must supply the appropriate cable(s) to mate the system under test to the I/O panel.

Appendix A contains a bill of materials for the major components of the present AIDS mechanization. The fabrication of the computing subsystem was quickly achieved using an industrial chassis incorporating a 12-slot card cage and integral power supply. Minor modifications to the chassis control panel were required to provide for a PROM set select switch, a bus timeout monitor indicator, and several test points. These additions are interfaced to the computing subsystem via circuitry on the universal prototype board.

The various computing subsystem boards listed in appendix A are I/O mapped as shown in figure 4 and memory mapped as shown in figure 5. The central processor unit (CPU) board contains an 8085A microprocessor, which provides adequate computational capability for currently planned operating modes. Table 1 shows the assignment of system interrupts.

The floppy diskette drive unit is a dual-drive single density standard sized diskette system. It interfaces directly to the floppy diskette controller board in the computing subsystem. The single density format provides more than ample storage capability. One drive is used for system program modules, and the other is used for scratch file storage.

The operator terminal is a single unit with a full sized black and white CRT screen and full keyboard. The CRT and keyboard are interfaced to the computing subsystem via a full duplex serial port on system expansion board A. High speed refresh of the CRT display is performed in vectored cursor mode at 1920 characters per second. A minor terminal modification was necessary to provide software control over the cursor marker on the screen. This was achieved by rewiring the keyboard enable/disable flip-flop, which is under software control, to the cursor blanking circuit. This allows the cursor to be blanked during screen refresh operations, resulting in a flicker-free display. The keyboard has been wired permanently enabled.

The line printer is a 5 by 7 dot matrix printer with a dual channel vertical forms unit that allows the proper pagination of all system printouts. The interface to the computing subsystem is via a parallel discretes port on the central processor board.

The I/O panel is a NASA-designed and -constructed unit which provides the user an interface with the computing subsystem for analog and discrete signals. Figure 6 shows the signal paths within the I/O panel. The connectors for the user interface cable(s) are located on the rear of the AIDS cabinet. For each discrete, monitoring jacks and light-emitting diode (LED) indicator lamps are provided on the front of the I/O panel. Internal to the I/O panel are line drivers and receivers for the discretes, which provide the user with a balanced differential double-rail interface. The receivers interface to the computing subsystem via system expansion board A, and the drivers interface via system expansion board B. With regard to analog trunks, the I/O panel is passive and provides only breakout jacks on the front panel. The analog inputs interface with the computing subsystem via the analog input board, which scans the inputs using a ±10 volt balanced multiplexer. The ±10 volt unbalanced analog outputs from the computing subsystem are fed from the four analog output boards.
Figure 3. Aircraft interrogation and display system.
Figure 4. AIDS I/O address map.
Figure 5. AIDS memory map.
### TABLE 1.—AIDS INTERRUPT ALLOCATION

<table>
<thead>
<tr>
<th>Level</th>
<th>Assignment</th>
<th>Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trap</td>
<td>Not used</td>
<td>----</td>
</tr>
<tr>
<td>A</td>
<td>Bus timeout</td>
<td>AIDS tally only</td>
</tr>
<tr>
<td>B</td>
<td>Not used</td>
<td>----</td>
</tr>
<tr>
<td>C</td>
<td>Not used</td>
<td>----</td>
</tr>
<tr>
<td>0</td>
<td>INTR pushbutton</td>
<td>User manual interrupt</td>
</tr>
<tr>
<td>1</td>
<td>Timer no. 0</td>
<td>RTMTX task wait timer</td>
</tr>
<tr>
<td>2</td>
<td>Disk controller</td>
<td>RTMTX diskette I/O</td>
</tr>
<tr>
<td>3</td>
<td>Timer no. 1</td>
<td>User clock</td>
</tr>
<tr>
<td>4</td>
<td>External interrupt</td>
<td>User sync</td>
</tr>
<tr>
<td>5</td>
<td>1 Hz interrupt</td>
<td>AIDS time of day clock</td>
</tr>
<tr>
<td>6</td>
<td>USART C receiver</td>
<td>RTMTX terminal handler</td>
</tr>
<tr>
<td>7</td>
<td>USART C transmitter</td>
<td>RTMTX terminal handler</td>
</tr>
</tbody>
</table>
Figure 6. AIDS I/O paths.
SOFTWARE DESIGN

Two separate software systems are resident within the AIDS. They are alternately accessible to the operator via a PROM select switch on the front of the computing subsystem chassis. One system is the maintenance and diagnostic software system, which consists of a commercial monitor package designed for the central processor board plus a NASA-designed set of extension routines. This package, which is stored as firmware on two PROMs that are installed on the central processor board, is executed when the PROM select switch is in the "monitor" position. This software provides basic AIDS troubleshooting services and diskette subsystem test routines.

The second software system is the main AIDS hierarchy, which consists of the components shown in figure 7. This software structure is shown from bottom to top in the order the four components become active in the system. The first component to execute is the RTMTX, which is a commercial package designed to be used with the central processor board and provides diskette subsystem services. This package is stored as firmware on eight PROMs installed on the system expansion boards and is executed when the PROM select switch is in the "disk" position. The remaining three software components are loaded into the AIDS memory from the system diskette in drive 0, and are mapped as shown in figure 8.

Embedded in the RTMTX firmware is a configuration module that defines the characteristics and mapping of the diskette subsystem hardware. It also specifies the tasks to be created when the system is initialized. The task list includes the diskette drive controller board handler, the diskette I/O handler, several diskette directory servicing routines, the full terminal handler, and the bootstrap loader. These routines and associated variables are accessible via PUBLIC labels, which can be linked to user code. Since the RTMTX code requires no maintenance, the PROM set never requires reprogramming and the integrity of the hardware is enhanced. Appendix B contains a listing of the configuration module and the SUBMIT file used to create the RTMTX firmware.

When the AIDS is powered up (or reset) with the PROM select switch in the "disk" position, the RTMTX begins executing and sets up the tasks specified by the configuration module. When the bootstrap loader becomes the active task, it seeks a file called RMXSYS on the system diskette, loads it into random access memory, and starts executing it. The file : F0: RMXSYS always contains the AIDS supervisor task module component of the AIDS software hierarchy. Once loaded, this module assumes central control of the system and is the point to which all other components return when execution is completed.

The AIDS supervisor contains an initialization routine followed by a looping command interpreter routine. It also contains many routines which are commonly needed by the different AIDS users. These include the CRT/KB handler, printer handler, analog I/O drivers, scratch diskette librarian, time-of-day clock, display data formatters, and utility routines. These can be accessed by a user via hard-mapped linkages in the common data area.

One of the functions performed by the AIDS supervisor during the initialization procedure is to request the RTMTX to load a module called USER from the system diskette. The file : F0: USER always contains the user main module component of the software hierarchy. Within it are contained the user interrupt servicing routines, user I/O packages, and an initialization subroutine. It also contains tables defining the syntax for user commands and user scratch file load control.
Figure 7. AIDS software hierarchy.
Figure 8. AIDS RAM allocation.
The fourth component of the AIDS software hierarchy are overlays. Overlay modules are generally loaded and executed in response to a keyboard command, and they always provide a specific function. They are linked to the remaining software via absolute entry addresses within the overlay area and, like the USER main module, have access to AIDS supervisor subroutines and variables via the common area. In general, each overlay has associated with it a unique display which is presented on the CRT. Overlays may be either system or user related. Most are operator interactive, and all must exit back to the AIDS supervisor when the KB escape key is pressed. System overlays provide functions such as interrupt control, printer moding, clock management, and I/O panel monitoring. User overlays are not restricted as to function but must conform to the mapping, linkage, and escape conventions required of all AIDS overlays.

Taken as a whole, the design of the AIDS software is intended to provide a multitasking environment within which the various system and user tasks can share a single CPU. The lowest priority task is always the servicing of the operator interface, which includes the CRT, KB, and printer. All higher priority tasks are invoked by interrupts, which require temporarily halting the operator I/O. A typical user application might involve responding to a sync interrupt from the system under test, inputting data, performing computations, outputting data, and setting up a data buffer for the current operator display. As the amount of time required to service such an interrupt increases, the most noticeable effect is the slowing of CRT screen refresh. Another variable that affects screen refresh is the amount of data being displayed, since there is computational overhead associated with formatting as well as screen write operations. The performance of the AIDS in various applications will be later quantified as a duty cycle or percentage of time which is devoted to interrupt-driven code execution as opposed to operator I/O.

USER EXPERIENCE

Since 1978, the AIDS has been employed in support of three research projects and is planned for use in at least two others. Two AIDS units are in active use, and a third unit is soon to enter service. The F-8 DFBW iron bird application (ref. 1) allows closed-loop aerodynamic simulation and redundant sensor fault emulation, providing valuable support in software verification and validation. The HIMAT remotely piloted research vehicle application (ref. 2) provides open-loop display of onboard computer memory data. It is used extensively in support of simulation, preflight testing, and system troubleshooting. Another user project is an experimental nodal network data bus breadboard (ref. 3). For this project the AIDS provides test set capability for the I/O processor in each node and monitors bus message traffic. A planned future application is support for the AFTI/F-111 project (ref. 4) where the AIDS will monitor the interchannel message traffic within the redundant flight system. Another future application is support for the DAST ARW-II project (ref. 5), where the AIDS will provide test set capability for a multiprocessor flight computer as well as provide the usual data display functions.

One measure of the performance of the AIDS is the loading or level of saturation of its central processor for each application. Loading may be defined as the duty cycle or percentage of time required to perform time-critical (interrupt-linked) computational tasks as opposed to operator I/O functions. The duty cycle ranges from 90 percent for the F-8 DFBW simulation to 10 percent for the HIMAT data display function. Screen refresh rates for the F-8 DFBW are very low (typically 0.5 per second). For a typical HIMAT display, however, the refresh rate is comfortably high (4 per second). The time required to perform a line printer hard copy of a display snapshot is roughly proportional to the refresh rate of the display and varies from 20 seconds to 5 seconds.

16
The HiMAT application best demonstrates the capabilities of the AIDS, and it has accumulated the most AIDS operating time, with over 2000 hours in a 3 year period. This application grew out of the need to augment the data display capability of the manufacturer-supplied GSE, called the system test console (STC). The STC mates with the HiMAT aircraft umbilical connector, and one of its functions is to allow the contents of the onboard computer memory to be examined. However, the STC can only display a single byte as a bit pattern expressed in octal digits, severely limiting the visibility of the functioning of the onboard computer.

To provide the needed additional display capability, the AIDS was connected to the STC as shown in figure 9. The 16 address lines are tied in common to the STC thumbwheels used for manual RAM address selection. The 8 data lines are tied to the output from the onboard computer, which feeds the decoders driving the STC octal display. The AIDS sequentially outputs an address, waits for a sync pulse from the onboard computer, and then reads the RAM data byte output by the computer. This sequence is repeated every 20 milliseconds, which is the rate at which the onboard program services the test console interface.

The AIDS operator controls which addresses are to be read by creating with KB inputs a formatted CRT display (called a page) that specifies by data type and RAM memory location, which items are desired. Table 2 shows the different data display formats available to the operator. Of these, only codes VG and DG (specially scaled fixed-point formats for the vertical gyro and directional gyro, respectively) are unique to HiMAT. Note that a single data item causes from 1 to 15 successive RAM addresses to be read. The AIDS software builds an address table based on the display requirements and scans this table repetitively. As the data is returned, it is buffered, formatted for display, and presented on the CRT in a continuously refreshed mode.

Appendix C contains hard copies of representative HiMAT displays. Also shown is a typical scratch diskette directory page and a hard copy of the command interpreter display, which lists the system and user commands available. The HiMAT project uses these display pages and others to support software verification and validation, system maintenance, preflight and postflight tests, and closed-loop simulations. Over 100 display page formats of various types have been created and placed on scratch diskette. The AIDS has become an integral part of such critical testing as the preflight test, where AIDS data dumps are written into several procedure sequences. The ability to select a scratch diskette and quickly (in 1 to 3 seconds) load any of up to 45 display page files has been of great benefit to the HiMAT project. In addition, the inherent flexibility of the software system has been demonstrated repeatedly by the changes that have easily been implemented in response to project engineering request.

CONCLUDING REMARKS

General purpose user-programmable ground support equipment has been developed and placed in service in support of both aircraft and simulation facilities. Three years' experience involving several users has demonstrated the utility of the system concept and created a demand for additional systems to support future users. The flexibility of the concept has been demonstrated in a wide range of applications, including real-time data acquisition, software verification and validation, system integration testing, and real-time closed-loop simulation.

The major contribution of the system, known as the aircraft interrogation and display system (AIDS), has been its ability to make visible the functioning of a digital flight
Figure 9. AIDS to HiMAT interface.
### Table 2: AIDS Data Display Formats

<table>
<thead>
<tr>
<th>Number</th>
<th>Code</th>
<th>Number of bytes</th>
<th>Number of bits</th>
<th>Data type</th>
<th>Sign?</th>
<th>Number of columns</th>
<th>Display format</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>H1</td>
<td>1</td>
<td>8</td>
<td>Any</td>
<td>---</td>
<td>2</td>
<td>HH</td>
</tr>
<tr>
<td>2</td>
<td>H2</td>
<td>2</td>
<td>16</td>
<td>Any</td>
<td>---</td>
<td>4</td>
<td>HHHH</td>
</tr>
<tr>
<td>3</td>
<td>H3</td>
<td>3</td>
<td>24</td>
<td>Any</td>
<td>---</td>
<td>6</td>
<td>HHHHHHHH</td>
</tr>
<tr>
<td>4</td>
<td>H4</td>
<td>4</td>
<td>32</td>
<td>Any</td>
<td>---</td>
<td>8</td>
<td>HHHHHHHHHH</td>
</tr>
<tr>
<td>5</td>
<td>I1</td>
<td>1</td>
<td>8</td>
<td>Any</td>
<td>---</td>
<td>8</td>
<td>BBBBBBB</td>
</tr>
<tr>
<td>6</td>
<td>I2</td>
<td>2</td>
<td>16</td>
<td>Any</td>
<td>---</td>
<td>16</td>
<td>BBBBBBBBBBBBBBB</td>
</tr>
<tr>
<td>7</td>
<td>F4</td>
<td>4</td>
<td>32</td>
<td>Floating point</td>
<td>Y</td>
<td>10</td>
<td>[ - ]DDDD.DDDD</td>
</tr>
<tr>
<td>8</td>
<td>I1</td>
<td>1</td>
<td>8</td>
<td>Integer</td>
<td>N</td>
<td>4</td>
<td>_DDD</td>
</tr>
<tr>
<td>9</td>
<td>I2</td>
<td>2</td>
<td>16</td>
<td>Integer</td>
<td>N</td>
<td>6</td>
<td>_DDDDDD</td>
</tr>
<tr>
<td>10</td>
<td>D1</td>
<td>1</td>
<td>8</td>
<td>Integer</td>
<td>Y</td>
<td>4</td>
<td>[ - ]DDDD</td>
</tr>
<tr>
<td>11</td>
<td>D2</td>
<td>2</td>
<td>16</td>
<td>Integer</td>
<td>Y</td>
<td>6</td>
<td>[ - ]DDDDDD</td>
</tr>
<tr>
<td>12</td>
<td>D3</td>
<td>3</td>
<td>24</td>
<td>DAC value</td>
<td>Y</td>
<td>6</td>
<td>HHH</td>
</tr>
<tr>
<td>13</td>
<td>D4</td>
<td>4</td>
<td>32</td>
<td>DAC value</td>
<td>Y</td>
<td>7</td>
<td>[ - ]DD.DDD</td>
</tr>
<tr>
<td>14</td>
<td>D5</td>
<td>5</td>
<td>40</td>
<td>DAC value</td>
<td>Y</td>
<td>8</td>
<td>OOO</td>
</tr>
<tr>
<td>15</td>
<td>O1</td>
<td>1</td>
<td>8</td>
<td>Any</td>
<td>---</td>
<td>3</td>
<td>OOO</td>
</tr>
<tr>
<td>16</td>
<td>O2</td>
<td>2</td>
<td>16</td>
<td>Any</td>
<td>---</td>
<td>6</td>
<td>OOOOOO</td>
</tr>
<tr>
<td>17</td>
<td>A1</td>
<td>1</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>1</td>
<td>A</td>
</tr>
<tr>
<td>18</td>
<td>A2</td>
<td>2</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>2</td>
<td>AA</td>
</tr>
<tr>
<td>19</td>
<td>A3</td>
<td>3</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>3</td>
<td>AAA</td>
</tr>
<tr>
<td>20</td>
<td>A4</td>
<td>4</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>4</td>
<td>AAAA</td>
</tr>
<tr>
<td>21</td>
<td>A5</td>
<td>5</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>5</td>
<td>AAAAA</td>
</tr>
<tr>
<td>22</td>
<td>A6</td>
<td>6</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>6</td>
<td>AAAAAA</td>
</tr>
<tr>
<td>23</td>
<td>A7</td>
<td>7</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>7</td>
<td>AAAAAAAA</td>
</tr>
<tr>
<td>24</td>
<td>A8</td>
<td>8</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>8</td>
<td>AAAAAAAAAA</td>
</tr>
<tr>
<td>25</td>
<td>A9</td>
<td>9</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>9</td>
<td>AAAAAAAAA</td>
</tr>
<tr>
<td>26</td>
<td>AA</td>
<td>10</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>10</td>
<td>AAAAAAAAAA</td>
</tr>
<tr>
<td>27</td>
<td>AB</td>
<td>11</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>11</td>
<td>AAAAAAAAAAAA</td>
</tr>
<tr>
<td>28</td>
<td>AC</td>
<td>12</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>12</td>
<td>AAAAAAAAAAAAAA</td>
</tr>
<tr>
<td>29</td>
<td>AD</td>
<td>13</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>13</td>
<td>AAAAAAAAAA</td>
</tr>
<tr>
<td>30</td>
<td>AE</td>
<td>14</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>14</td>
<td>AAAAAAAAA</td>
</tr>
<tr>
<td>31</td>
<td>AF</td>
<td>15</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>15</td>
<td>AAAAAAAAA</td>
</tr>
<tr>
<td>32</td>
<td>E0</td>
<td>1</td>
<td>1</td>
<td>Event bit 0</td>
<td>---</td>
<td>4</td>
<td>&quot;ONE&quot; or &quot;ZERO&quot;</td>
</tr>
<tr>
<td>33</td>
<td>E1</td>
<td>1</td>
<td>1</td>
<td>Event bit 1</td>
<td>---</td>
<td>4</td>
<td>&quot;ONE&quot; or &quot;ZERO&quot;</td>
</tr>
<tr>
<td>34</td>
<td>E2</td>
<td>1</td>
<td>1</td>
<td>Event bit 2</td>
<td>---</td>
<td>4</td>
<td>&quot;ONE&quot; or &quot;ZERO&quot;</td>
</tr>
<tr>
<td>35</td>
<td>E3</td>
<td>1</td>
<td>1</td>
<td>Event bit 3</td>
<td>---</td>
<td>4</td>
<td>&quot;ONE&quot; or &quot;ZERO&quot;</td>
</tr>
<tr>
<td>36</td>
<td>E4</td>
<td>1</td>
<td>1</td>
<td>Event bit 4</td>
<td>---</td>
<td>4</td>
<td>&quot;ONE&quot; or &quot;ZERO&quot;</td>
</tr>
<tr>
<td>37</td>
<td>E5</td>
<td>1</td>
<td>1</td>
<td>Event bit 5</td>
<td>---</td>
<td>4</td>
<td>&quot;ONE&quot; or &quot;ZERO&quot;</td>
</tr>
<tr>
<td>38</td>
<td>E6</td>
<td>1</td>
<td>1</td>
<td>Event bit 6</td>
<td>---</td>
<td>4</td>
<td>&quot;ONE&quot; or &quot;ZERO&quot;</td>
</tr>
<tr>
<td>39</td>
<td>E7</td>
<td>1</td>
<td>1</td>
<td>Event bit 7</td>
<td>---</td>
<td>4</td>
<td>&quot;ONE&quot; or &quot;ZERO&quot;</td>
</tr>
<tr>
<td>40</td>
<td>F1</td>
<td>1</td>
<td>8</td>
<td>Fixed point</td>
<td>Y</td>
<td>10</td>
<td>[ - ]DDDD.DDD</td>
</tr>
<tr>
<td>41</td>
<td>F2</td>
<td>2</td>
<td>16</td>
<td>Fixed point</td>
<td>Y</td>
<td>10</td>
<td>[ - ]DDDD.DDD</td>
</tr>
<tr>
<td>42</td>
<td>DG</td>
<td>2</td>
<td>16</td>
<td>Directional gyro</td>
<td>Y</td>
<td>10</td>
<td>[ - ]DDDD.DDD</td>
</tr>
<tr>
<td>43</td>
<td>VG</td>
<td>2</td>
<td>16</td>
<td>Vertical gyro</td>
<td>Y</td>
<td>10</td>
<td>[ - ]DDDD.DDD</td>
</tr>
</tbody>
</table>

Display format key:
- **H** = hexadecimal digit 0 to 9, A to F
- **B** = binary digit 0 or 1
- **D** = decimal digit 0 to 9
- **O** = octal digit 0 to 7
- **A** = any ASCII character
system, thus enhancing test coverage, troubleshooting, and the efficiency with which experiments are conducted.

The use of off-the-shelf commercial hardware and operating system software greatly reduced the development effort and cost of ownership.

Because of the capabilities of AIDS and its user-oriented operational features, experience to date, which has involved a complex flight development and integration project, has been excellent, with extremely high acceptance.

National Aeronautics and Space Administration
Ames Research Center
Dryden Flight Research Facility
February 3, 1982
APPENDIX A.—AIDS COMPONENTS

This appendix lists the components of AIDS.
The major computing subsystem components, which are from the Intel Corporation, are as follows:

ICS-80 KIT 640 Chassis and Power Supply (1 each)
   Rack mount chassis, control panel, heavy duty power supply, four-slot card cage module, multibus backplane

SBC 614 Card Cage Modules (2 each)
   Expands above kit to 12 slots capacity

SBC 80/30 Central Processor Board (1 each)
   8085A CPU, 16K bytes RAM, 4K bytes PROM, serial port, 24 discrete I/O lines, interval timer, interrupt controller

SBC 116 Expansion Boards (2 each)
   16K bytes RAM, 8K bytes PROM, 48 discretes I/O, serial port

SBC 724 Analog Output Boards (4 each)
   Each board provides four 12-bit DAC channels, range ±10 volts

SBC 711 Analog Input Board (1 each)
   Provides 16 balanced channels, range ±10 volts, 12-bit A/D

SBC 204 Floppy Diskette Controller Board (1 each)
   Provides control of two single-density standard sized drives

SBC 310 High Speed Math Unit Board (1 each)
   Provides 16-bit and 32-bit arithmetic, fixed and floating point

SBC 905 Universal Prototype Board (1 each)
   1 Hz clock circuitry, bus timeout monitor circuit, PROM switching control logic, external interrupt termination

RMX80 Real-Time Multitasking Executive (1 each)
   RMX830.LIB, BOT830.LIB, BOTUNR.LIB, DFSDIR.LIB, DIO830.LIB, DFSUNR.LIB, THI830.LIB, THO830.LIB, PLM80.LIB

Additional components of the AIDS are as follows:

Floppy Diskette Drive Unit
   Manufacturer: Data Systems Design, Inc.
   Type: DSD-110-IN-2A drive unit (1 each)
   DSD-CM chassis mount for rack (1 each)
   Interface: Cable provided to mate with SBC-204 controller
   Characteristics: Dual drives, standard sized floppy diskettes, single density IBM soft-sectored

Operator Terminal
   Manufacturer: SOROC Technology
   Type: IQ-120
   Interface: RS-232C serial
   Characteristics: 19,200 baud rate, 24 lines by 80 columns, vectored cursor capability
Line Printer
Manufacturer: Centronics Data Computer Corp.
Type: 306C
Interface: Standard Centronics parallel TTL interface
Characteristics: 5 X 7 dot matrix, tractor feed, 80/132 character/line, 120 character/second print rate, two-channel vertical forms unit
APPENDIX B. —AIDS REAL-TIME MULTITASKING EXECUTIVE LISTINGS

Following are printer listings generated during the building of the AIDS real-time multitasking executive firmware.
Configuration Module

This listing shows the software components which together comprise the software system create table. It defines the initial task table, the initial exchange table, several hardware definition tables, and miscellaneous data storage area declarations.

ASM80 :FI:CONFIG

ISIS-IT 8080/8085 MACRO ASSEMBLER, V3.0 CONFIG PAGE 1
AIDS RMX SYSTEM CONFIGURATION MODULE 12 DEC 1979

LOC OBJ LINE SOURCE STATEMENT

1 $ TITLE('AIDS RMX SYSTEM CONFIGURATION MODULE 12 DEC 1979')
 2 3 NAME CONFIG; R GLOVER
 4 PUBLIC ROCRTB
 5 PUBLIC ROL2EX,ROL3EX,ROL4EX,ROL5EX,RQLBEX,ROLAEX,ROLCEX
 6 7 CSEG
 8
 0000 0600 C 9 ROCRTB: DW ITT ; INITIAL TASK TABLE
 0002 0A 10 DB 10
 0003 000 C 11 DW IET ; INITIAL EXCHANGE TABLE
 0005 12 DB 18
 13 14 ITT:
 16 } DISK CONTROLLER TASK
 17 18 PUBLIC ROL2EX,CNTLIX
 19 EXTRN RQHD4
 20
 0006 53424332 21 TASK1: DB 'SRC204' ; DISK CONTROLLER BOARD HANDLER
 0008 3034
 000C 0000 E 22 DW RQHD4
 000E 9600 D 23 DW STK1
 0010 5000 24 DW 80
 0012 21 25 DB 33 ; INTERRUPT LEVEL 2 USED FOR 204 BOARD
 0013 7B00 D 26 DW CNTLIX
 0015 E002 D 27 DW TDI
 28 29 ; TERMINAL HANDLER TASK
 30 31 PUBLIC UREADX,WRITX
 32 EXTRN ROTHI,RQINX,RQOUTX,RQWAKE,RQDBUG,RQALRM,RQALAX,RQLEX
 33
 0017 5445524D 34 TASK2: DB 'TERMD'
 001B 494F
 001D 0000 E 35 DW ROTHI
 001F E000 D 36 DW STK2
 0021 2400 37 DW 36
 0023 70 38 DB 112 ; INTERRUPT LEVEL 6 USED FOR KEYBOARD INPUT
 0024 0000 E 39 DW RQOUTX
 0026 FE02 D 40 DW T92
 41 42 ; DISK I/O MAIN TASK
 43 44 EXTRN RQPSK,RQSSK
 45
 0028 4445524D 46 TASK3: DB 'DISKID'
 002C 494F
 0030 0000 E 47 DW RQPSK
 0034 0001 D 48 DW STK3
 0032 3000 49 DW 48
 0034 81 50 DB 129
 0035 0000 E 51 DW RQSSK

26
LOC OBJ LINE SOURCE STATEMENT

0037 1203 D 52 DM TB3

53

54 ; DISK SERVICES TASKS (6)

55

56 PUBLIC RDUBF;RDAB

57 EXTRN RDPDIR;RPATR,RPDEL,RDPFMT,RDPB,RDPAM

58 EXTRN RDIRX,ROATRX,RODELX,ROFMTX,ROBLX,ROANMX

59

0039 44455253 60 TASK4: DB 'DIRSVC' ; DIRECTORY SERVICES

003F 5643

0040 0000 E 61 DM RDPDIR

0041 3A01 D 62 DM STK4

0043 3000 D 63 DM 48

0045 82 64 DB 130

0046 0000 E 65 DM RDPDIR

0048 2603 D 66 DM TB4

004A 41545452 DD TASK5: DB 'ATTRIB' ; ATTRIBUTES

004E 4942

0050 0000 E 69 DM RDATR

0052 6A01 D 70 DM STK5

0054 4000 D 71 DM 64

0056 83 72 DB 131

0057 0000 E 73 DM RDATRX

0059 3A03 D 74 DM TB5

005B 44454A43 76 TASK6: DB 'DELETE'

005F 5445

0061 0000 E 77 DM RPDDEL

0063 A401 D 78 DM STK6

0065 4000 D 79 DM 64

0067 84 80 DB 132

0068 0000 E 81 DM RDDELX

006A 4E03 D 82 DM TB6

006C 464A524D 84 TASK7: DB 'FORMAT'

0070 4154

0072 0000 E 85 DM RDPFMT

0074 3A01 D 86 DM STK7

0076 4000 D 87 DM 64

0078 85 88 DB 133

0079 0000 E 89 DM RDPFMTX

007B 6203 D 90 DM TB7

007D 4C4F3444 92 TASK8: DB 'LOAD'

0081 2E20

0083 0000 E 93 DM RDPBL

0085 2402 D 94 DM STK8

0087 4000 95 DM 64

0089 86 96 DB 134

008A 0000 E 97 DM RDPBL

008C 7603 D 98 DM TB8

99

009E 52454E41 100 TASK9: DB 'RENAME'

009F 4D45
LOC  OBJ  LINE  SOURCE STATEMENT

0094 0000  E 101  DW  ROPRM
0096 4A02  D 102  DW  STKP
0098 4000  103  DW  64
009A 87  104  DW  135
009B 0000  E 105  DW  ROPRM
009D BA03  D 106  DW  TD9
107
108 ; BOOTSTRAP LOADER TASK
109
110  PUBLIC ROPOOL
111  EXTRM ROBOOT
112
009F 424F4534  113 TASK10: DB  'BOOT ' 114 00A3 2020
00A5 0000  E 114  DW  ROBOOT
00A7 A02  115  DW  STK10
00A8 4000  116  DW  64
00A9 FE  117  DB  254
00AC 0000  118  DW  0
00AE 9E03  D 119  DW  TD10
120
121 ; INITIAL EXCHANGE TABLE
122
00BD 0F00  123 IET: DW  ROL2EX
00BE 7000  124  DW  CNTLIX
00B4 0000  E 125  DW  RQINPX
00B6 0000  E 126  DW  RQOUTX
00B8 0000  E 127  DW  RQHAME
00BA 0000  E 128  DW  RQBUG
00BC 0000  E 129  DW  RQARM
00BE 0000  E 130  DW  RQLEX
00C0 0000  E 131  DW  RQLEX
00C2 8200  132  DW  UREADX
00C4 B000  133  DW  WMRTX
00C6 0000  E 134  DW  RQDSKX
00C8 0000  E 135  DW  RQDIRX
00C4 0000  E 136  DW  RQATRX
00C2 0000  E 137  DW  RQDELX
00CE 0000  E 138  DW  RQMTX
00D0 0000  E 139  DW  RQLOX
00D2 0000  E 140  DW  RQRMHX
141
142 ; TABLES FOR DISK CONTROLLER TASK
143
144  PUBLIC ROCS: RQDEV: RQDCT: RQDC4
145
00D4 02  146 ROCS: DB  2  ; CONTROLLER SPECIFICATION TABLE
00D5 70  147 DB  70H  ; I/O ADDRESS
00D6 02  148 DB  2  ; INTERRUPT LEVEL 2
00E7 0F00  D 149  DW  ROL2EX
00E9 7000  D 150  DW  CNTLIX
151
00DB 02  152 RQDEV: DB  2  ; NUMBER OF DRIVES
153
00DC 4630  154 RQDCT: DB  'FD'  ; DEVICE CONFIGURATION TABLE
<table>
<thead>
<tr>
<th>LOC</th>
<th>OBJ</th>
<th>LINE</th>
<th>SOURCE STATEMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>00E0</td>
<td>00</td>
<td>155</td>
<td>DB 0:0:0</td>
</tr>
<tr>
<td>00E0</td>
<td>00</td>
<td>156</td>
<td>DB 'F1'</td>
</tr>
<tr>
<td>00E0</td>
<td>00</td>
<td>157</td>
<td>DB 0:0:1</td>
</tr>
<tr>
<td>00E0</td>
<td>00</td>
<td>158</td>
<td></td>
</tr>
<tr>
<td>00E0</td>
<td>00</td>
<td>159</td>
<td>RQDC4: DB 1 ; DRIVE CHARACTERISTICS TABLE</td>
</tr>
<tr>
<td>00E7</td>
<td>70</td>
<td>160</td>
<td>DB 70H ; 204 BOARD I/O ADDRESS</td>
</tr>
<tr>
<td>00E8</td>
<td>00</td>
<td>161</td>
<td>DB 0 ; CONTROLLER CHIP 0</td>
</tr>
<tr>
<td>00E9</td>
<td>00</td>
<td>162</td>
<td>DB 8 ; TRACK STEP TIME = 8 MS</td>
</tr>
<tr>
<td>00EA</td>
<td>00</td>
<td>163</td>
<td>DB 8 ; HEAD SETTLING TIME = 8 MS</td>
</tr>
<tr>
<td>00EB</td>
<td>49</td>
<td>164</td>
<td>DB 49H ; INDEX COUNT = 4 , LOAD TIME = 35 MS</td>
</tr>
<tr>
<td>00EC</td>
<td>00</td>
<td>165</td>
<td>; BUFFER ALLOCATION BLOCK</td>
</tr>
<tr>
<td>00ED</td>
<td>00</td>
<td>166</td>
<td>DB 0 ; STATIC MODE</td>
</tr>
<tr>
<td>00EE</td>
<td>00</td>
<td>167</td>
<td>DB 0 ; MAXIMUM OF 2 FILES CONCURRENTLY OPEN</td>
</tr>
<tr>
<td>00F0</td>
<td>00</td>
<td>168</td>
<td>DB 2 ; MEMORY</td>
</tr>
<tr>
<td>00F1</td>
<td>00</td>
<td>169</td>
<td>DB 8 ; BARRAM</td>
</tr>
<tr>
<td>00F2</td>
<td>00</td>
<td>170</td>
<td>DB 8 ; STK1: DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>00F3</td>
<td>00</td>
<td>171</td>
<td></td>
</tr>
<tr>
<td>00F4</td>
<td>00</td>
<td>172</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>00F5</td>
<td>00</td>
<td>173</td>
<td></td>
</tr>
<tr>
<td>00F6</td>
<td>00</td>
<td>174</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>00F7</td>
<td>00</td>
<td>175</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>00F8</td>
<td>00</td>
<td>176</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>00F9</td>
<td>00</td>
<td>177</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>00FA</td>
<td>00</td>
<td>178</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>00FB</td>
<td>00</td>
<td>179</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>00FC</td>
<td>00</td>
<td>180</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>00FD</td>
<td>00</td>
<td>181</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>00FE</td>
<td>00</td>
<td>182</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>00FF</td>
<td>00</td>
<td>183</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0100</td>
<td>00</td>
<td>184</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0101</td>
<td>00</td>
<td>185</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0102</td>
<td>00</td>
<td>186</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0103</td>
<td>00</td>
<td>187</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0104</td>
<td>00</td>
<td>188</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0105</td>
<td>00</td>
<td>189</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0106</td>
<td>00</td>
<td>190</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0107</td>
<td>00</td>
<td>191</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0108</td>
<td>00</td>
<td>192</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0109</td>
<td>00</td>
<td>193</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>010A</td>
<td>00</td>
<td>194</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>010B</td>
<td>00</td>
<td>195</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>010C</td>
<td>00</td>
<td>196</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>010D</td>
<td>00</td>
<td>197</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>010E</td>
<td>00</td>
<td>198</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>010F</td>
<td>00</td>
<td>199</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0110</td>
<td>00</td>
<td>200</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0111</td>
<td>00</td>
<td>201</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0112</td>
<td>00</td>
<td>202</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0113</td>
<td>00</td>
<td>203</td>
<td>DS 80 ; STACK AREA</td>
</tr>
<tr>
<td>0114</td>
<td>00</td>
<td>204</td>
<td>DS 80 ; STACK AREA</td>
</tr>
</tbody>
</table>

LOC | OBJ | LINE | SOURCE STATEMENT
--- | --- | --- | ---
038A | 205 | TBP: | DS 20
037E | 206 | TB10: | DS 20

0382 | 208 | RQPOOL: | DS 256 J BOOTSTRAP LOADER BUFFER
0482 | 209 | RQBBUF: | DS 700 J DIRSV BUFFER
076E | 210 | BABRAM: | DS 800 J RAB BUFFER

211
212 END

PUBLIC SYMBOLS
CNTLIX D 007B RQAB C 00EC RQCRTB C 0000 RQDST C 0004 RQDBUF D 04B2 RQCT D 002C RQBARA C 0046
RQLOEX D 0000 RQLEX D 000F RQLEX D 001E RQLEX D 002D RQLEX D 003C RQLEX D 004B RQLEX D 005A
RQLEX D 0069 RQNEVE D 0038 RQPOOL D 0382 UREADX D 0082 UNRITX D 008C

EXTERNAL SYMBOLS
RQALRM E 0000 RQATRX E 0000 RQBBUF E 0000 RQBBUF E 0000 RQDBEX E 0000 RQDIRX E 0000 RQDDXX E 0000
RQMTX E 0000 RQMTX E 0000 RQMTX E 0000 RQMTX E 0000 RQMTX E 0000 RQMTX E 0000 RQMTX E 0000
RQPATR E 0000 RQBPUL E 0000 RQBPUL E 0000 RQBPUL E 0000 RQBPUL E 0000 RQBPUL E 0000 RQBPUL E 0000
RQRAMX E 0000 RQTHDI E 0000 RQTHDI E 0000 RQTHDI E 0000 RQTHDI E 0000 RQTHDI E 0000 RQTHDI E 0000

USER SYMBOLS
BABRAM D 076E CNTLIX D 007B IET C 0080 ITT C 0064 RDQSN E 0000 RDQSN E 0000 RDQSN E 0000 RDQSN E 0000
RQBOOT E 0000 RQCRTB C 0000 RQDST C 0004 RQDFU D 04B2 RQDBUF E 0000 RQDCT C 008C RQDCT C 008C
RQDIRX E 0000 RQDRS E 0000 RQDXX E 0000 RQDDXX E 0000 RQDDXX E 0000 RQDDXX E 0000 RQDDXX E 0000
RQLEX D 0000 RQLEX D 000F RQLEX D 001E RQLEX D 002D RQLEX D 003C RQLEX D 004B RQLEX D 005A
RQLEX D 0069 RQLEX D 0069 RQLEX D 0069 RQLEX D 0069 RQLEX D 0069 RQLEX D 0069 RQLEX D 0069
RQPUL E 0000 RQBPUL E 0000 RQBPUL E 0000 RQBPUL E 0000 RQBPUL E 0000 RQBPUL E 0000 RQBPUL E 0000
RQTHDI E 0000 RQSWAL E 0000 RQSWAL E 0000 RQSWAL E 0000 RQSWAL E 0000 RQSWAL E 0000 RQSWAL E 0000
STKS D 016A STKS D 016A STKS D 016A STKS D 016A STKS D 016A STKS D 016A STKS D 016A
TA$: 017 TASK2 C 002B TASK4 C 0039 TASK5 C 004A TASK6 C 0058 TASK7 C 0039 TASK8 C 0079
TASKC C 005E TD1 D 002A TD1 D 002A TD1 D 002A TD1 D 002A TD1 D 002A TD1 D 002A TD1 D 002A
TD6 D 034E TD6 D 034E TD6 D 034E TD6 D 034E TD6 D 034E TD6 D 034E TD6 D 034E

ASSEMBLY COMPLETE, NO ERRORS
This listing defines the sequence of operations performed by the MDS in building the firmware. The configuration module is linked with the other RTMTX modules, located at address 40H, and finally converted to a HEX file, which is used to program the PROMs.

```
LINK &
  :F1:BOT830.LIB(VECRST), &
  :F1:RMX830.LIB(START), &
  :F1:RMX830.LIB(SUSPND,RESUME,DLTASK,DLEXCH), &
  :F1:CONFIG.OBJ, &
  :F1:BOT830.LIB, &
  :F1:DFSDIR.LIB(SEEK,DIRECTORY,ATTRIB,DELETE,RENAME,LOAD), &
  :F1:DFSDIR.LIB(FORMAT,FORMAT201,FMTTABLE), &
  :F1:DI0830.LIB, &
  :F1:DFSUNR.LIB, &
  :F1:THI830.LIB, &
  :F1:TH0830.LIB, &
  :F1:RMX830.LIB, &
  :F1:BOTUNR.LIB, &
  :F1:PLM80.LIB &
    TO &
  :F1:ROM.OBJ &
    MAP &
    PRINT(:F1:ROMLNK.LST)
LOCATE &
  :F1:ROM.OBJ &
    TO &
  :F1:ROM.ABS &
    CODE(40H) &
    STACKSIZE(0) &
    DATA(ODC00H) &
    MAP &
    PUBLICS &
    SYMBOLS &
    LINES &
    PRINT(:F1:ROMLOC.LST)
ATTRIB :F1:ROM.HEX W0
DELETE :F1:ROM.HEX
OBJHEX :F1:ROM.ABS TO :F1:ROM.HEX
ATTRIB :F1:ROM.HEX W1
COPY :F1:ROMLNK.LST TO :LP:
COPY :F1:NAME TO :LP:
COPY :F1:DATE TO :LP:
COPY :F1:ROMLOC.LST TO :LP:
```
Linker Listing

This listing is generated by the object linker and provides a list of all modules included.

ISIS-II OBJECT LINKER V3.0 INVOKED BY:
  -LINK &
** :F1:BOT830.LIB(VECRST), &
** :F1:RMX830.LIB(START), &
** :F1:RMX830.LIB(SUSPND,RESUME,DLTASK,DLEXCH), &
** :F1:CONFIG.OBJ, &
** :F1:B0T830.LIB, &
** :F1:DFSDBR.LIB(SEEK,DIRECTORY,ATTRIB,DELETE,RENAME,LOAD), &
** :F1:DFSDBR.LIB(FORMAT,FORMAT201,FORMAT30), &
** :F1:DI0B30.LIB, &
** :F1:DF°UNR.LIB, &
** :F1:TH030.LIB, &
** :F1:TH830.LIB, &
** :F1:RMX830.LIB, &
** :F1:B0T830.LIB, &
** :F1:PLMB80.LIB &
** TO &
** :F1:ROM.OBJ &
** MAP &
** PRINT(:F1:ROMLNK.LST)

LINK MAP OF MODULE ROM
WRITTEN TO FILE :F1:ROM.OBJ
MODULE IS A MAIN MODULE

SEGMENT INFORMATION:
START STOP LENGTH REL NAME

    3EFBH  B   CODE
    1275H  B   DATA
       40H  B   STACK
0000H  0002H  3H  A  ABSOLUTE
0002H  000AH  3H  A  ABSOLUTE
0010H  0012H  3H  A  ABSOLUTE
0018H  001AH  3H  A  ABSOLUTE
0020H  0022H  3H  A  ABSOLUTE
0024H  002EH  9H  A  ABSOLUTE
0030H  0032H  3H  A  ABSOLUTE
0034H  0036H  3H  A  ABSOLUTE
0038H  003AH  3H  A  ABSOLUTE
003CH  003EH  3H  A  ABSOLUTE

INPUT MODULES INCLUDE:
  :F1:BOT830.LIB(VECRST)
  :F1:RMX830.LIB(START)
  :F1:RMX830.LIB(SUSPND)
  :F1:RMX830.LIB(RESUME)
  :F1:RMX830.LIB(DL TASK)
  :F1:RMX830.LIB(DLEXCH)
  :F1:CONFIG.OBJ(CONFIG)
  :F1:BOT830.LIB(IN830P)
  :F1:BOT830.LIB(ROB00T)
  :F1:BOT830.LIB(FILNAM)
  :F1:BOT830.LIB(RDSECT)
  :F1:DFSDBR.LIB(SEEK)
  :F1:DFSDBR.LIB(DIRECTORY)
  :F1:DFSDBR.LIB(ATTRIB)
  :F1:DFSDBR.LIB(DELETE)
  :F1:DFSDBR.LIB(REN AME)
  :F1:DFSDBR.LIB(LOAD)
  :F1:DFSDBR.LIB(FORMAT)
  :F1:DFSDBR.LIB(FORMAT201)
:FI:DFSDIR.LIB(FMTTABLE)
:FI:DI0B30.LIB(DISKIO)
:FI:DI0B30.LIB(HAN204)
:FI:DFSUNR.LIB(NOFORMAT202)
:FI:DFSUNR.LIB(NOFORMAT204)
:FI:DFSUNR.LIB(NOFORMAT206)
:FI:DFSUNR.LIB(DRIVETIMEOUTVAL)
:FI:DFSUNR.LIB(MINISTARTUP)
:FI:THI830.LIB(TDHINI)
:FI:THI830.LIB(ECHO)
:FI:THI830.LIB(CHKINP)
:FI:THI830.LIB(PRINP)
:FI:THI830.LIB(SCANBAUDRATE)
:FI:THI830.LIB(LNEDIT)
:FI:TH0830.LIB(THDINO)
:FI:TH0830.LIB(CONTROL)
:FI:TH0830.LIB(USART8030)
:FI:TH0830.LIB(CONTRCTRLTABLE)
:FI:TH0830.LIB(MERGER)
:FI:RMX830.LIB(SYNCH)
:FI:RMX830.LIB(DLYLST)
:FI:RMX830.LIB(DLYLIST)
:FI:RMX830.LIB(DIJASON)
:FI:RMX830.LIB(SL)
:FI:RMX830.LIB(RMSYLL)
:FI:RMX830.LIB(ENTSLL)
:FI:RMX830.LIB(TB8030)
:FI:BOTUNR.LIB(THRATE)
:FI:BOTUNR.LIB(RESETV)
:FI:BOTUNR.LIB(NODBGIR)
:FI:BOTUNR.LIB(FILUNR)
:FI:PLMB80.LIB(OP0011)
:FI:PLMB80.LIB(OP0014)
:FI:PLMB80.LIB(OP0018)
:FI:PLMB80.LIB(OP0025)
:FI:PLMB80.LIB(OP0029)
:FI:PLMB80.LIB(OP0034)
:FI:PLMB80.LIB(OP0086)
:FI:PLMB80.LIB(OP0091)
:FI:PLMB80.LIB(OP0094)
:FI:PLMB80.LIB(OP0096)
:FI:PLMB80.LIB(OP0098)
:FI:PLMB80.LIB(OP0101)
:FI:PLMB80.LIB(OP0103)
Locater Listing

This listing is generated by the object locater and provides a complete list of all PUBLIC symbols.

ISIS-II OBJECT LOCATER V3.0 INVOKE DBY:
~LOCATE &
** :F1:ROM.OBJ &
** TO &
** :F1:ROM.ABS &
** CODE(40H) &
** STACKSIZE(0) &
** DATA(O0COOH) &
** MAP &
** PUBLICS &
** SYMBOLS &
** LINES &
** PRINT(:F1:ROMLOC.LST)

SYMBOL TABLE OF MODULE ROM
READ FROM FILE :F1:ROM.OBJ
WRITTEN TO FILE :F1:ROM.ABS

VALUE TYPE SYMBOL

0000H PUB R?VECRST
0040H PUB R?GSTRT
00C1H PUB R?SUSP
00EFH PUB R?RESM
0110H PUB R?DTSK
0166H PUB R?DXCH
0271H PUB R?BAD
0185H PUB R?CRTB
0259H PUB R?CST
0261H PUB R?CCT
0268H PUB R?DC4
0260H PUB R?DEV
044EH PUB R?INTDI
03FFH PUB R?INTEI
037FH PUB R?INTINI
03E7H PUB R?MASK
0433H PUB R?CLVL
0448H PUB R?ELVL
0259H PUB R?ENDI
0469H PUB R?SETV
02F3H PUB R?UPFRI
0483H PUB R?BOOT
0615H PUB R?STSTR
061EH PUB R?Sект
0674H PUB R?SEEK
0ABAH PUB R?ETBLK
1095H PUB R?LSMAP
0A80H PUB R?MAIN
1590H PUB R?FOPENCHECK
188AH PUB R?DEL
1CE1H PUB R?FOPEN
1071H PUB R?BTDIR
1382H PUB R?ADERE
109FH PUB R?RFB
160BH PUB R?RVALIYREQUEST
120BH PUB R?PBREAD
149CH PUB R?FNAMECHECK
123CH PUB R?PTTSK
107EH PUB R?RSDIR
0E2BH PUB R?RIGET
0C82H PUB R?FREQB
1283H PUB R?DSAVE
0A0DH PUB R?ABS10
DCF2H PUB UREADX
DCFCH PUB UWRITX
E753H PUB R?ADRXCH
E75IH PUB R?INITM
E73FH PUB R?SIMVEC
E769H PUB R?RESPFX
E75FH PUB ROBOTX
E755H PUB RQLOADX
E775H PUB R?SLPMSG
E77EH PUB R?NAME
E76EH PUB R?CLSKIL
E7F0H PUB R?FBLORG
EBEFH PUB R?FCBLISTLOCK
E903H PUB R?BITMAPLOCK
E7FCH PUB R?RETURNBUFFXCH
EB00H PUB R?DISFTKSTD
EBF9H PUB R?DIRECTRORYLOCK
EB7AH PUB R?ABS1OM
E990H PUB ROATRX
E9C0H PUB RODELX
E9E1H PUB RORNMX
EA09H PUB ROLLX
EA48H PUB R°COMPX
EA7BH PUB R?ENTX1
EA7CH PUB R°BSKX
EA7AH PUB R?ENT204
EA95H PUB R°INPX
EA9FH PUB R°INSK
EA9CH PUB R°RNAME
EA9DH PUB R°ROATRX
E9COH PUB ROIC
EAE0H PUB R?CHARINPEXC
EAEAH PUB R?ECHOEXC
ED60H PUB R?ALARMSS
ED77H PUB R°CNTRL
ED81H PUB R?OUTX
ED86H PUB R?ALRM
ED95H PUB R?L7EX
EE08H PUB R???RL
EE0BH PUB R°ACTV
EE24H PUB R???EL
EE36H PUB R°LIEL
EE6OH PUB R???ELR
EE64H PUB R??TTLR
EE66H PUB R???SLR

MEMORY MAP OF MODULE ROM
READ FROM FILE :FI:ROM.OBJ
WRITTEN TO FILE :FI:ROM.ABS
MODULE START ADDRESS 0040H

START STOP LENGTH REL NAME

0000H 0002H 3H A ABSOLUTE
0008H 000AH 3H A ABSOLUTE
0010H 0012H 3H A ABSOLUTE
0018H 001AH 3H A ABSOLUTE
0020H 0022H 3H A ABSOLUTE
0024H 002EH 9H A ABSOLUTE
0030H 0032H 3H A ABSOLUTE
0034H 0036H 3H A ABSOLUTE
0038H 003AH 3H A ABSOLUTE
003CH 003EH 3H A ABSOLUTE
0040H 3F3AH 3EFBH B CODE
DC00H EE74H 1275H B DATA
EE75H F6BFH 04BH B MEMORY
APPENDIX C.—TYPICAL HiMAT DISPLAYS

This appendix describes some of the displays used in the HiMAT program.
Command Interpreter Display

This is the display to which the AIDS executive returns when the user has terminated the previous operation. This display provides the operator with the following information: (1) the version of the AIDS executive, (2) the name and version number of the user module, (3) a list of the available user commands, and (4) a list of the available system commands. The operator enters the desired command, and the corresponding overlay is loaded and executed. A special case is the command "LD" which is used to activate the displays stored on the scratch diskette: (1) the scratch diskette directory is examined to determine the page number of the file specified, (2) the corresponding overlay is loaded, (3) the display templates are copied from the scratch diskette file into the overlay, and (4) the display is activated in refreshed mode.

Hard copy of the HiMAT command interpreter display:

```
AIRCRAFT INTERROGATION & DISPLAY SYSTEM

AIDS-II SYSTEM EXECUTIVE  16 SEPT 1980  R GLOVER

USER LOAD MODULE NAME : HIMAT 8.15.08

USER COMMANDS :
FF  MP  MC  MD  MT  TX
A1  A2  A3

SYSTEM COMMANDS :
IC  TC  PC  DK  LD  FD
SIO  SMP  SMS  SMD  SMT
```

Scratch Diskette Directory Display

This display is generated by the AIDS executive in response to a "DK" command. It shows the name of the scratch diskette currently in drive number 1 and lists the contents of each of the 45 available files. The operator has a menu of commands to choose from:

- LD = load a file and present the display in refreshed mode
- SAVE = write the current overlay display into a selected file
- INIT = initialize a new scratch diskette with selected name
- DEL = delete a selected file
- NAME = rename a selected file
Tabular Data Display

This display is accessed by the user command "MP" and allows the user to define a display of up to 20 data items. For each item the user must specify item number, data type, hexadecimal address, description, and units. In addition, if the data type is either F1 or F2, the operator must also enter the zero and maximum scaling of the parameter in engineering units. Once created, the display may be saved on the scratch diskette if desired.
Block Memory Dump Data Display

This page format is accessed by command "MD" and allows the operator to display in hexadecimal format up to 304 bytes in a single block. The operator must specify the beginning and ending addresses of the block. The display may be saved on scratch diskette file if desired.

Hard copy of typical HiMAT block memory dump display:

```
OPERATE
HIMAT 8.15.88

PAGE 5 MEMORY DUMP

61D7 40 00 00 00 33 00 00 C7 30 00 01 00 00 00 00 00
61E7 92 04 00 00 88 00 01 00 00 00 00 00 48 00 00 00
61F7 00 40 11 00 00 00 E0 00 40 00 06 00 1C 40 00
```

DISK : HIMAT G. P. 1 FILE NO. 42 FILE NAME : C FAIL 1

Free-Form Data Display

This display mode is accessed by the user command "FF" and allows the operator to create unstructured displays in any format desired. Separate commands are available to allow creating the static or background portion of the display, followed by the insertion of data items in any desired format at any location of the screen. Once created, the display may be saved on scratch diskette if desired.
Hard copy of typical HiMAT free-form data display:

```
OPERATE
HIMAT 8.15.80

ENGINE PANEL

- 14.422 (PSI).
- COMPRESSOR
- DISCHARGE
- PRESSURE

- PLAD = 15.000 DEG
- PLAC = 0.000 DEG

- 10.000 (%) 0.000 (C) 99.996 (%) 0.000 (DEG)
- RPM
- EXHAUST
- GAS
- NOZZLE
- TEMPERATURE
- AREA

- CONTROL
- ENGINE
- NOZZLE
- MODE
- STABILITY
- CONTROL
- IGNITION
- COMBAT
- HIGH
- OVERRIDE
- ZERO

DISK : HIMAT G. P. 1 FILE NO. 31 FILE NAME : ENG PANEL

REFERENCES


```
A microprocessor-based general purpose ground support equipment for electronic systems has been developed and placed in service at the NASA Dryden Flight Research Facility. The hardware and software are designed to permit diverse applications in support of aircraft flight systems and simulation facilities. This paper describes the implementation of the hardware and the structure of the software and describes the application of the system to an ongoing research aircraft project.