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
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:

```
System under test \rightarrow Data \rightarrow AIDS display formatter \rightarrow Refreshed data display
```

Open-loop function generator:

```
AIDS \rightarrow Real-time stimuli \rightarrow Digital or analog computer
```

Redundant sensor simulator:

```
Simplex sensor variable \rightarrow AIDS \rightarrow Simulated redundant sensor signals \rightarrow Redundant digital flight control system
Sensor fault parameters
```

Closed-loop simulation:

```
AIDS aircraft model \rightarrow Model outputs \rightarrow Flight control system under test \rightarrow System commands
```

Figure 1. Examples of conceptual AIDS applications.
User-created software modules

System diskette files

Scratch diskette files

Real-time multitasking executive

AIDS supervisor

Operator I/O

User application supervisor

User timing control

User I/O servicing

Flight system under test

Analog recording

Control keyboard

CRT data display

Display hard copy

User-programmed clock

Sync

Figure 2. AIDS functional overview.
HARDWARE DESCRIPTION

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.
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>
<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>HHHHHHHHHH</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>B1</td>
<td>1</td>
<td>8</td>
<td>Any</td>
<td>---</td>
<td>8</td>
<td>BBBBBBBB</td>
</tr>
<tr>
<td>6</td>
<td>H2</td>
<td>2</td>
<td>16</td>
<td>Any</td>
<td>---</td>
<td>16</td>
<td>BBBBBBBBHBHBHBHBHBHBHB</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.DDD</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>_DDDD</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>_DDDD</td>
</tr>
<tr>
<td>12</td>
<td>DD</td>
<td>2</td>
<td>12</td>
<td>DAC value</td>
<td>Y</td>
<td>3</td>
<td>HHHH</td>
</tr>
<tr>
<td>13</td>
<td>DH</td>
<td>2</td>
<td>12</td>
<td>DAC value</td>
<td>Y</td>
<td>7</td>
<td>[ - ]DD.DD</td>
</tr>
<tr>
<td>14</td>
<td>DV</td>
<td>2</td>
<td>12</td>
<td>DAC value</td>
<td>Y</td>
<td>10</td>
<td>[ - ]DDDD.DDD</td>
</tr>
<tr>
<td>15</td>
<td>O1</td>
<td>1</td>
<td>8</td>
<td>Any</td>
<td>---</td>
<td>3</td>
<td>000</td>
</tr>
<tr>
<td>16</td>
<td>O2</td>
<td>2</td>
<td>16</td>
<td>Any</td>
<td>---</td>
<td>6</td>
<td>000000000</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>_AAAAAAA</td>
</tr>
<tr>
<td>24</td>
<td>A8</td>
<td>8</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>8</td>
<td>_AAAAAA</td>
</tr>
<tr>
<td>25</td>
<td>A9</td>
<td>9</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>9</td>
<td>_AAAAAAAA</td>
</tr>
<tr>
<td>26</td>
<td>AA</td>
<td>10</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>10</td>
<td>_AAAAAAAAAAAAAAAA</td>
</tr>
<tr>
<td>27</td>
<td>AB</td>
<td>11</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>11</td>
<td>_AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA</td>
</tr>
<tr>
<td>28</td>
<td>AC</td>
<td>12</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>12</td>
<td>_AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA</td>
</tr>
<tr>
<td>29</td>
<td>AD</td>
<td>13</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>13</td>
<td>_AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA</td>
</tr>
<tr>
<td>30</td>
<td>AE</td>
<td>14</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>14</td>
<td>_AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA</td>
</tr>
<tr>
<td>31</td>
<td>AF</td>
<td>15</td>
<td>---</td>
<td>ASCII string</td>
<td>---</td>
<td>15</td>
<td>_AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA</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 discrete 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, PLMS0.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
<table>
<thead>
<tr>
<th>Line Printer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manufacturer:</td>
</tr>
<tr>
<td>Centronics Data Computer Corp.</td>
</tr>
<tr>
<td>Type:</td>
</tr>
<tr>
<td>306C</td>
</tr>
<tr>
<td>Interface:</td>
</tr>
<tr>
<td>Standard Centronics parallel TTL interface</td>
</tr>
<tr>
<td>Characteristics:</td>
</tr>
<tr>
<td>5 X 7 dot matrix, tractor feed, 80/132 character/line, 120 character/second print rate, two-channel vertical forms unit</td>
</tr>
</tbody>
</table>
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 RNX SYSTEM CONFIGURATION MODULE 12 DEC 1979

LOC OBJ LINE SOURCE STATEMENT

I $ TITLE('AIDS RNX SYSTEM CONFIGURATION MODULE 12 DEC 1979')
2 NAME CONFIG ; R Glover
4 PUBLIC ROCRITB
5 PUBLIC ROLDEX,ROLDEX,ROLDEX,ROLDEX,ROLBEX,ROLBEX
6 CSEG
8
0000 0600 C 9 ROCRITB; DW ITT ; INITIAL TASK TABLE
0002 0A 10 DB 10
0003 0B00 C 11 DW IET ; INITIAL EXCHANGE TABLE
0005 12 DB 18

14 ITT: DISK CONTROL TASK
15 PUBLIC ROL2EX,CNTLIX
16 EXTRN RQHD4
20
0006 53423332 21 TASK1: DB 'SRC204' ; DISK CONTROLLER BOARD HANDLER
000A 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 7000 26 DW CNTLIX
0015 E202 27 DW TDI
28
29 ; TERMINAL HANDLER TASK
30
31 PUBLIC UREADX,WRITRX
32 EXTRN ROTH1,RQINPX,RQOUTX,ROWAKE,RQBUG,RQALRM,ROLAEX,ROLTEX
33
34 TASK2: DB 'TERMD' 34
001B 494F
001D 0000 E 35 DW ROTH1
001F E200 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 E202 40 DW TDI
41
42 ; DISK I/O MAIN TASK
43
44 EXTRN RQDSK,RQDSKX
45
46 TASK3: DB 'DISKID' 46
002B 444534B
002E 494F
0030 0000 E 47 DW RQDSK
0032 0000 D 48 DW STK3
0032 0000 49 DW 48
0034 81 50 DB 129
0035 0000 E 51 DW RQDSKX

26
I IS IS BOBO BOBO BO BOBO MACRO ASSEMBLER, V3.0 CONFIG PAGE 2

AIDS RMX SYSTEM CONFIGURATION MODULE 12 DEC 1979

LOC OBJ LINE SOURCE STATEMENT

0037 1203 D 52 DW TB3
53
54; DISK SERVICES TASKS (6)
55
56 PUBLIC RQDBUF;RQDB
57 EXTRN RQDIR;RQPAT;RQPDEL;RQPFMT;RQPL;RQPAN
58 EXTRN RQDIR;RQATRX;RQDELX;RQPFMT;RQBL;RQANMK
59

0039 44455253 60 TASK4: DB 'DIRSVC' ; DIRECTORY SERVICES
0039 5642
003F 0000 E 61 DW RQDIR
0041 3A01 D 62 DW STK4
0043 3000 63 DW 48
0045 82 64 DB 130
0046 0000 E 65 DW RQDIRX
0049 2603 D 66 DW TB4
004A 41545452 67 TASK5: DB 'ATTRIB' ; ATTRIBUTES
004E 4942
0050 0000 E 69 DW RQPATR
0052 6041 D 70 DW STK5
0054 4000 71 DW 64
0056 83 72 DB 131
0057 0000 E 73 DW RQDIRX
0059 3A03 D 74 DW TB5
005A 44454C45 75 TASK6: DB 'DELETE'
005F 5445
0061 0000 E 77 DW RQPDEL
0063 6041 D 78 DW STK6
0065 4000 79 DW 64
0067 84 80 DB 132
0068 0000 E 81 DW RQDELX
006A 4E03 D 82 DW TB6
006B 4445524D 83 TASK7: DB 'FORMAT'
006C 4154
0070 4154
0072 0000 E 85 DW RQPFMT
0074 6A01 D 86 DW STK7
0076 4000 87 DW 64
0078 85 88 DB 133
0079 0000 E 89 DW RQPFMT
007B 6203 D 90 DW TB7
0081 4A4F4E41 91 TASK8: DB 'LOAD'
0081 2620
0083 0000 E 93 DW RQPLD
0085 2A02 D 94 DW STK8
0087 4000 95 DW 64
0089 86 96 DB 134
008A 0000 E 97 DW RQPL
008C 7603 D 98 DW TB8
008E 52445441 99 TASK9: DB 'RENAME'
0092 4045

27
LOC  OBJ  LINE  SOURCE STATEMENT

AIDS RMX SYSTEM CONFIGURATION MODULE  12 DEC 1979

LOC  OBJ  LINE  SOURCE STATEMENT

0094 0000  E   101  DW  ROPRM
0096 6A02  D   102  DW  STKP
0098 4000  D   103  DW  D4
009A 87    D   104  DB  135
009B 0000  E   105  DW  ROPRM
009D BA03  D   106  DW  TDP
009F 424F4F5A 113 TASK10:  DB  'BOOT'
00A3 2020
00A7 0000  E   114  DW  ROBOOT
00A7 A02   D   115  DW  STK10
00A8 4000  D   116  DW  D4
00A8 FE    D   117  DB  254
00AC 0000  D   118  DW  0
00AE 9E03  D   119  DW  TD10
00B0 0F00  D   123  JET:  DW  ROL2EX
154  DB  70H ; 204 BOARD I/O ADDRESS
00B2 7000  D   124  DW  CNTLIX
00B4 0000  E   125  DW  RQINPX
00B6 0000  E   126  DW  RQOUTX
00B8 0000  E   127  DW  RQHAXE
00BA 0000  E   128  DW  RQBUG
00BC 0000  E   129  DW  SQALRM
00BE 0000  E   130  DW  ROLEX
00C0 0000  E   131  DW  RQLEX
00C2 8200  D   132  DW  UREAXX
00C4 8C00  D   133  DW  UARTX
00C6 0000  E   134  DW  RQDSKX
00C8 0000  E   135  DW  RQDIRX
00CA 0000  E   136  DW  RQATRX
00CC 0000  E   137  DW  RQRELX
00CE 0000  E   138  DW  RQRMX
00D0 0000  E   139  DW  RQDXL
00D2 0000  E   140  DW  RQRMX
142  / TABLES FOR DISK CONTROLLER TASK
143  PUBLIC ROCST,ROQDEV,ROQCT,ROOCT4
145  DB  2  CONTROLLER SPECIFICATION TABLE
146  DB  2H  I/O ADDRESS
147  DB  2  INTERRUPT LEVEL 2
148  DB  2  NUMBER OF DRIVES
149  DW  ROL2EX
150  DW  CNTLIX
151  DW  ROL2EX
152  'FO'  DEVICE CONFIGURATION TABLE
154  DW  ROL2EX
LOC OBJ LINE SOURCE STATEMENT

008 00  132  DB  0:0:0
009 00
00E 00
00E 453I  156  DB  'F1'
00E 00  157  DB  0:0:1
00E 00
00E 01
00E 01  159  RGBDC4: DB  1 ; DRIVE CHARACTERISTICS TABLE
00E 70  160  DB  71H ; 204 BOARD I/O ADDRESS
00E 00  161  DB  0 ; CONTROLLER CHIP 0
00E 00  162  DB  B ; TRACK STEP TIME = 8 MS
00E 0B  163  DB  B ; HEAD SETTLING TIME = 8 MS
00E 49  164  DB  49H ; INDEX COUNT = 4 ; LOAD TIME = 35 MS
165
166 ; BUFFER ALLOCATION BLOCK
167
00E 0000  168  RGBAB: DW  0:0 ; STATIC MODE
00E 0000
00F 02  169  DB  2 ; MAXIMUM OF 2 FILES CONCURRENTLY OPEN
00F 4657  170  DW  BARRAM
171
172  DBEG
173
0000  174  RGBEX: DS  15 ; EXCHANGE AREA
000F  175  RGBEX: DS  15
001E  176  RGBEX: DS  15
0020  177  RGBEX: DS  15
0022  178  RGBEX: DS  15
0040  179  RGBEX: DS  15
0058  180  RGBEX: DS  15
0067  181  RGBEX: DS  15
0070  182  CMTLIX: DS  10
0082  183  VREADX: DS  10
008C  184  VWRITE: DS  10
185
0096  186  STK1: DS  80 ; STACK AREA
0066  187  STK2: DS  36
010A  188  STK3: DS  48
013A  189  STK4: DS  48
016A  190  STK5: DS  64
01A0  191  STK6: DS  64
01A8  192  STK7: DS  64
022A  193  STK8: DS  64
026A  194  STKY: DS  64
029A  195  STK10: DS  64
196
02EA  197  TB1: DS  20 ; TASK DESCRIPTOR AREA
02FE  198  TB2: DS  20
031E  199  TB3: DS  20
036A  200  TB4: DS  20
033A  201  TB5: DS  20
03AE  202  TB6: DS  20
0362  203  TB7: DS  20
0576  204  TB8: DS  20

29
LOC OBJ LINE SOURCE STATEMENT

03BA 205 TB9: DS 20
03FE 206 TB10: DS 20
207
0382 208 RQPOOL: DS 256 ; BOOTSTRAP LOADER BUFFER
0482 209 RQBUFS: DS 700 ; DISK SVC BUFFER
0782 210 BRRAM: DS 800 ; RAS BUFFER
211
212 END

PUBLIC SYMBOLS

CNTLIX D 007B RQBA C 006C RQCDTB C 0000 RQDST C 0004 RQDBUF D 0482 RQDCALL C 006C RQBAR C 006C
RQLOEX D 0000 RQLEX D 000F RQLEX D 001E RQLEX D 002B RQLEX D 003C RQLEX D 004B RQLEX D 005A
RQLEX D 0069 RQNEV C 002B RQPOOL D 03B2 ; RQPOOL D 03B2 ; RQPOOL D 03B2

EXTERNAL SYMBOLS

RQALRM E 0000 RQATRX E 0000 RQBOOT E 0000 RQBUG E 0000 RQDELX E 0000 RQDIRX E 0000 RQDSKX E 0000
RQFMFX E 0000 RQHDA E 0000 RQINPX E 0000 RQLEX E 0000 RQLEX E 0000 RQLEX E 0000 RQLEX E 0000
RQPALE E 0000 RQPEFIL E 0000 RQPALE E 0000 RQPALE E 0000 RQPALE E 0000 RQPALE E 0000
RQPEH X E 0000 RQTHD I E 0000 RQTHD E 0000

USER SYMBOLS

BRRAM D 0782 CNTLIX D 007B IET C 0000 IIT C 0006 RQALRM E 0000 RQATRX E 0000 RQBAR C 006C
RQBOOT E 0000 RQCDTB C 0000 RQDST C 0004 RQDBUF D 0482 RQDBUF E 0000 RQDCALL C 006C RQBAR C 006C
RQDIRX E 0000 RQDIRX E 0000 RQDIRX E 0000 RQDIRX E 0000 RQDIRX E 0000 RQDIRX E 0000 RQDIRX E 0000
RQLEX D 000F RQLEX D 001E RQLEX D 002B RQLEX D 003C RQLEX D 004B RQLEX D 005A RQLEX D 0069
RQLEX D 007B RQLEX D 008B RQLEX D 009D RQLEX D 00A6 RQLEX D 00B9 RQLEX D 00CB RQLEX D 00DC
RQPALE E 0000 RQPALE E 0000 RQPALE E 0000 RQPALE E 0000 RQPALE E 0000 RQPALE E 0000 RQPALE E 0000
RQPALE E 0000 RQPALE E 0000 RQPALE E 0000 RQPALE E 0000 RQPALE E 0000 RQPALE E 0000 RQPALE E 0000
RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000
RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000
RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000
RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000 RQTHD I E 0000

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.

```plaintext
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:THO830.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(ODC0OH) &
    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 &
*  LINK &
** :F1:BOT830.LIB(VECRST), &
** :F1:RMX830.LIB(START), &
** :F1:RMX830.LIB(SUSPND,RESUME,DLTASK,DLLEXCH), &
** :F1:CONFIG.OBJ, &
** :F1:IN830.LIB, &
** :F1:IN830.LIB(SUSF,NI],
** :F1:RMX830.LIB (RESUME),
** :F1:RMX830.LIB (DLTASK),
** :F1:RMX830.LIB (DLEXCH),
** :F1:RMSUNR.LIB, &
** :F1:THI830.LIB, &
** :F1:IN830.LIB, &
** :F1:IN830.LIB (RESUME),
** :F1:RMSUNR.LIB, &
** :F1:PLMB80.LIB &
** :F1:RMX830.LIB &
** :F1:ROMLNK,LST &
** 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

<table>
<thead>
<tr>
<th>Address</th>
<th>Length</th>
<th>Type</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000H</td>
<td>1275H</td>
<td>CODE</td>
<td>3EFBH</td>
</tr>
<tr>
<td>0000H</td>
<td>0002H</td>
<td>DATA</td>
<td>1275H</td>
</tr>
<tr>
<td>0000H</td>
<td>0010H</td>
<td>STACK</td>
<td>40H</td>
</tr>
<tr>
<td>0001H</td>
<td>0012H</td>
<td></td>
<td>0H</td>
</tr>
<tr>
<td>0001H</td>
<td>001AH</td>
<td></td>
<td>3H</td>
</tr>
<tr>
<td>0002H</td>
<td>0022H</td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>0002H</td>
<td>002EH</td>
<td></td>
<td>3H</td>
</tr>
<tr>
<td>0003H</td>
<td>0032H</td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>0003H</td>
<td>0036H</td>
<td></td>
<td>3H</td>
</tr>
<tr>
<td>0003H</td>
<td>003AH</td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>0003H</td>
<td>003EH</td>
<td></td>
<td>3H</td>
</tr>
</tbody>
</table>

INPUT MODULES INCLUDED:
:F1:BOT830.LIB(VECRST)
:F1:RMX830.LIB(START)
:F1:RMX830.LIB(SUSPND)
:F1:RMX830.LIB(RESUME)
:F1:RMX830.LIB(DLLEXCH)
:F1:RMX830.LIB(LOAD)
:F1:IN830.LIB(IN830P)
:F1:BOT830.LIB(GDBO00T)
:F1:IN830.LIB(FILNAM)
:F1:IN830.LIB(RDSECT)
:F1:DFSDIR.LIB(SEEK)
:F1:DFSDIR.LIB(DIRECTORY)
:F1:DFSDIR.LIB(ATTRIB)
:F1:DFSDIR.LIB(DELETE)
:F1:DFSDIR.LIB(RENAME)
:F1:DFSDIR.LIB(LOAD)
:F1:DFSDIR.LIB(FORMAT)
:F1:DFSDIR.LIB(FORMAT201)
:F1:DFSISR.LIB(FMTTABLE)
:F1:DIO830.LIB(DISKIO)
:F1:DIO830.LIB(HAN204)
:F1:DFSUNR.LIB(NOFORMAT202)
:F1:DFSUNR.LIB(NOFORMAT204)
:F1:DFSUNR.LIB(NOFORMAT206)
:F1:DFSUNR.LIB(DRIVETIMEOUT)
:F1:DFSUNR.LIB(MINISTARTUP)
:F1:THIG30.LIB(THDINI)
:F1:THIG30.LIB(ECHO)
:F1:THIG30.LIB(STDINP)
:F1:THIG30.LIB(PRINP)
:F1:THIG30.LIB(SCANBAUDRATE)
:F1:THIG30.LIB(LNEDIT)
:F1:THG30.LIB(THDINO)
:F1:THG30.LIB(CONTROL)
:F1:THG30.LIB(USART030)
:F1:THG30.LIB(CNTRLTABLE)
:F1:THG30.LIB(MERGER)
:F1:RMX830.LIB(SYNCH)
:F1:RMX830.LIB(RDYLST)
:F1:RMX830.LIB(DLYLST)
:F1:RMX830.LIB(DBJMAN)
:F1:RMX830.LIB(SL)
:F1:RMX830.LIB(RMVSSL)
:F1:RMX830.LIB(ENTSLL)
:F1:RMX830.LIB(TBB030)
:F1:BOTUNR.LIB(THRATE)
:F1:BOTUNR.LIB(RESETV)
:F1:BOTUNR.LIB(NODBGR)
:F1:BOTUNR.LIB(FILUNR)
:F1:PLM80.LIB(@P0011)
:F1:PLM80.LIB(@P0014)
:F1:PLM80.LIB(@P0018)
:F1:PLM80.LIB(@P0025)
:F1:PLM80.LIB(@P0029)
:F1:PLM80.LIB(@P0034)
:F1:PLM80.LIB(@P0086)
:F1:PLM80.LIB(@P0091)
:F1:PLM80.LIB(@P0094)
:F1:PLM80.LIB(@P0096)
:F1:PLM80.LIB(@P0098)
:F1:PLM80.LIB(@P0101)
:F1:PLM80.LIB(@P0103)
Locator Listing

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

ISIS-II OBJECT LOCATER V3.0 INVOKED BY:
-LOCATE &
** :F1:ROM.OBJ &
** TO &
** :F1:ROM.ABS &
** CODE(40H) &
** STACKSIZE(0) &
** DATA(0DC0OH) &
** 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?BSTR
0061H PUB R?SUSP
006FH PUB R?RESM
011DH PUB R?DTSK
0166H PUB R?DXCH
0271H PUB R?BAD
0185H PUB R?CRTB
0259H PUB R?CST
0261H PUB R?CT
0268H PUB R?DRC4
0260H PUB R?DEV
044EH PUB R?INTDI
03FFH PUB R?INTEI
037FH PUB R?INTINI
03E7H PUB R?LMASK
0463H PUB R?DLVL
0448H PUB R?ELVL
0285H PUB R?ENDI
0469H PUB R?SETV
023FH PUB R?DFPRI
0483H PUB R?BOOT
0615H PUB R?BSTR
061EH PUB R?DSECT
0674H PUB R?ISEEK
0A0AH PUB R?ETBLK
1095H PUB R?LSMAP
0A80H PUB R?MAPDBP
159DH PUB R?FILEOPENCHECK
1B8AH PUB R???DEL
1CE1H PUB R?POFN
1071H PUB R?BTDIR
1382H PUB R?MONOF
109FH PUB R?BTFCB
160BH PUB R?VALIDATEREQUEST
1282H PUB R?BREAD
149CH PUB R?FILENAMECHECK
123DH PUB R?DTTSK
107EH PUB R?RLSDIR
0E28H PUB R?DIRGET
0C22H PUB R?FREKB
1283H PUB R?DSAVE
0A0DH PUB R?ABS10
DCF2H PUB UREADX
DCFCH PUB UWRITX
E753H PUB R?ADRDXCH
E751H PUB R?INITCH
E73FH PUB R?SIMVEC
E769H PUB R?RESPEC
E75FH PUB R?ROBOTX
E755H PUB R?QLODIX
E775H PUB R?SLPMSG
E77EH PUB R?NAME
E766H PUB R?CLSKIL
E772H PUB R?FREEBUFXCH
E792H PUB R?RQPDX
E78CH PUB R?QIRX
E7F0H PUB R?QFLORG
E86FH PUB R?FCBLISTLOCK
E903H PUB R?BITMAPLOCK
E7FCUB PUB R?RETURNBUFCH
EBD8H PUB R?DISFTSTD
EBF9H PUB R?DIRECTORIESLOCK
EB7AH PUB R?ABS1OM
E99DH PUB R?QATUX
E7C0H PUB R?QDELX
E7E1H PUB R?QRNMX
EA09H PUB R?QDLX
EA4BH PUB R?QFMTX
EA7BH PUB R?QENTXI
EA7CH PUB R?QDSKX
EA7AH PUB R?QENT204
EA95H PUB R?QFINPX
EA9FH PUB R?QWAKE
EAA9H PUB R?Q6EX
EAB8H PUB R?QBUG
EAC2H PUB R?QINES
EAC8H PUB R?QINES
EAE6H PUB R?QCHARSS
EAEOH PUB R?QCHARFEXEC
EAEAH PUB R?QEOEXEC
ED6DH PUB R?QALARMSS
ED77H PUB R?QCTRL
ED81H PUB R?QOUTX
ED89H PUB R?QAUXRX
ED95H PUB R?RL7EX
EE08H PUB R?Q???RLR
EE0BH PUB R?QACTV
EE24H PUB R?Q???DLH
EE36H PUB R?QLIEX
EE6OH PUB R?Q???ERL
EE44H PUB R?Q???TLR
EE6CH PUB R?Q???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 84BH 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 8.15.00

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

SYSTEM COMMANDS :
IC TC PC DK LD FD
SIO SMP SMS SHD 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.

Hard copy of typical HiMAT page 3 display:

<table>
<thead>
<tr>
<th>ITEM</th>
<th>TYPE</th>
<th>ADDR</th>
<th>ZERO (EU)</th>
<th>MAX (EU)</th>
<th>DESCRIPTION</th>
<th>VALUE</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PITCH</td>
<td>158.798</td>
<td>DEGREES</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>6125</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>6131</td>
<td></td>
<td></td>
<td>ROLL</td>
<td>148.776</td>
<td>DEGREES</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>6188</td>
<td></td>
<td></td>
<td>YAW</td>
<td>331.260</td>
<td>DEGREES</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>6131</td>
<td></td>
<td></td>
<td>90 D ROLL</td>
<td>ZERO</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>6290</td>
<td></td>
<td></td>
<td>SPECIAL OP</td>
<td>-0.620</td>
<td>VOLTS</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>6109</td>
<td></td>
<td></td>
<td>UMB NOT SE</td>
<td>ONE</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>6109</td>
<td></td>
<td></td>
<td>LAN NOT SE</td>
<td>ONE</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td>6194</td>
<td>180.000</td>
<td>100.000</td>
<td>MIDV PR</td>
<td>0.098</td>
<td>DEG/SEC</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td>6195</td>
<td>180.000</td>
<td>100.000</td>
<td>MIDV YR</td>
<td>-0.049</td>
<td>DEG/SEC</td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
<td>6107</td>
<td></td>
<td></td>
<td>FAST ERECT</td>
<td>ZERO</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td>6109</td>
<td></td>
<td></td>
<td>PRI MODE</td>
<td>ZERO</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

DISK: HIRMAT G. P. 1  FILE NO. 6  FILE NAME: ATTITUDES
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 40 48 00 00 00
61F7  00 4D 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:

<table>
<thead>
<tr>
<th>OPERATE</th>
<th>PAGE 1</th>
<th>08:21:12</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIMAT 8.15.80</td>
<td>7.27.81</td>
<td></td>
</tr>
</tbody>
</table>

**ENGINE PANEL**

- COMPRESSOR: PLAD = 15.000 DEG
- DISCHARGE
- PRESSURE: PLAC = 0.000 DEG
- 18.000 (PSI)
- 14.422 (PSI)
- 10.000 (PSI)
- 0.000 (C)
- 99.996 (PSI)
- 0.000 (DEG)
- RPM
- EXHAUST
- EXHAUST
- THROTTLE
- GAS
- NOZZLE
- REG
- TEMPERATURE
- AREA
- CONTROL
- ENGINE
- NOZZLE
- MODE
- STABILITY
- CONTROL
- IGNITION
- COMBAT
- HIGH
- OVERRIDE
- ZERO
- ZER0
- ZER0
- ZER0

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.