CIO AFI DIAGNOSTIC DAD

External Reference Specification



Roseville Networks Division

Location Code: 52-7560

Project Number: 27114-60101

October 2, 1989

Version 2.1

Project Manager John McHugh

Project Engineer Bill Hooper



Roseville Networks Division

* HP Confidential *

Copyright © 1989 HEWLETT-PACKARD COMPANY

NOTICE

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

This document contains proprietary information which is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced or translated to another language without the prior written consent of Hewlett-Packard Company.

Copyright © 1988 by HEWLETT-PACKARD COMPANY

PRINTING HISTORY

New editions are complete revisions of the manual. Update packages, which are issued between editions, contain additional and replacement pages to be merged into the manual by the customer. The dates on the title page change only when a new edition or a new update is published. No information is incorporated into a reprinting unless it appears as a prior update; the edition does not change when an update is incorporated.

The software code printed alongside the date indicates the version level of the software product at the time the manual or update was issued. Many product updates and fixes do not require manual changes and, conversely, manual corrections may be done without accompanying product changes. Therefore, do not expect a one to one correspondence between product updates and manual updates.

First Edition		•		•		•	•							Unknov	wn
Revision			•	•			•					•		. OCT	88
Revision	•			•	•		•			•	•	•	•	AUG	89

LIST OF EFFECTIVE PAGES

The List of Effective Pages gives the date of the most recent version of each page in the manual. To verify that your manual contains the most current information, check the dates printed at the bottom of each page with those listed below. The date on the bottom of each page reflects the edition or subsequent update in which that page was printed.

Effective Pages	Date
All	 Aug 1989

Table of Contents

Section 1 INTRODUCTION

1.1 Project Identification	••		•••		•	 •	 	•		•	 •	 •		•		•	•			•	1-1
1.2 Product Team	••				•	 •	 	•			 •			•		•		• •		•	1-1
1.3 Abstract	• •			•	•		 				 •			•	• •	•				•	1-2
1.3.1 Field Support's Diagnostic Concerns	• •	••			•	 •	 	• •			 •					•					1-2
1.3.2 Product Verification's Diagnostic Cond	ern	s.			•	 •	 	•			 •	 •		•					•		1-2
1.3.3 Defect Analysis's Diagnostic Concerns	• •	• •			•		 	• •	•	•	 •			•		•		• •		•	1-3
1.4 Metrics to be Used	••	• •				 •	 	• •					• •	•							1-3

Section 2 PRODUCT ENVIRONMENT

2.1 Hardware Requirements	-1
2.2 Software Requirements	-1
2.2.1 Software Compatibility	-1
2.3 DAD/DAR Requirements	-2
2.3.1 Read Device Adapter Register	-3
2.3.2 Write Device Adapter Register	-3
2.3.3 Reset Device Adapter	-3
2.3.4 Read Data	- 3
2.3.5 Write Data	-3
2.3.6 Setup for Interrupt	-4
2.3.7 Check for Interrupt	-4
2.3.8 CIO Poll status	-4
2.3.9 Get CIO Slot	-4
2.3.10 Identify DAM	-4
2.4 Diagnostic-DAR Interface	-4
2.5 DAR Status Returns	-5

Section 3 EXTERNAL SPECIFICATIONS

3.1 User Environments	1
2) Discreti Suttem Composition	î
5.2 Diagnostic System Composition	1
3.3 User Interface.	2
3.4 Auto-diagnostics	4
3.5 Functional Description of Sections	4
3.5.1 Section 1 More Help 3-:	5
3.5.2 Section 2 Reset	6
3.5.3 Section 3 Identify	7
3.5.4 Section 4 Hardware Test	8
3.5.5 Section 5 Hardware Test with Testhood 3-10	0
3.5.6 Section 6 Status	2
3.5.7 Section 7 Register Level Input/Output Transactions	4

Section 4 HELP MESSAGES

4.1 Message 20	- 1
4.2 Message 10001	-1
4. 3 Message 10002	-2
4. 4 Message 30	-2
4.5 Messages for Section Help	- 3
4. 5. 1 Set 6, message 1	- 3
4. 5. 2 Set 6, message 2	- 3
4. 5. 3 Set 6, message 3	- 3
4. 5. 4 Set 6, message 4	-4
4. 5. 5 Set 6, message 5	-4
4. 5. 6 Set 6, message 6	-4
4. 5. 7 Set 6, message 7	-4

INTRODUCTION



1.1 PROJECT IDENTIFICATION

Product NameAsynchronous FIFO Interface DiagnosticProject MnemonicAFIDADPart Number27114-13601VersionExternal Reference Specification: Version 2.1

1.2 PRODUCT TEAM

Project ManagerJohn McHughDiagnostic Development EngineerBill HooperHardware Development EngineerThang Le
Pery PearsonProduct EngineerBill WangSupport EngineerDaryl WongSherlock ConvergenceBob Vaughan

* HP Confidential -- AFI Diagnostic ERS * 1-1

1.3 ABSTRACT

The Hewlett-Packard CIO AFI Diagnostic (henceforth referred to as AFIDAD) will be utilized by various groups of people, which include the following: 1) product verification personnel, 2) defect analysis personnel, and 3) field support personnel. Since each of these groups have their own definition of an "ideal" diagnostic, representatives of each group were interviewed to determine exactly what their interpretation of an "ideal" diagnostic is (i.e. what functionality requirements are *desired* by each group). The reasoning for querying these groups at the onset of the diagnostic development was so that each group's needs could be met as closely as possible to their "ideal" functionality requirements. The insight gained by the interviews is summarized below.

1.3.1 Field Support's Diagnostic Concerns

The primary concern of the field diagnostician is to isolate the faulty FRU(s) (field replaceable unit) as quickly as possible. However, the field diagnostician does not concern him/her-self with which component is defective, since component replacements usually are not done in the field. Therefore, as far as the field diagnostician is concerned, the "ideal" diagnostic is one which determines if the FRU(s) it tests is operational or defective as quickly as possible. The diagnostic need not attempt to isolate the fault to the component level (if a fault in fact exists), since this is an inefficient use of resources.

1.3.2 Product Verification's Diagnostic Concerns

The major concerns of the product verification personnel are as follows:

- A) Determine if the card in question functions properly. If a fault is in fact detected, it need not be isolated to the component level.
- B) The Tests should execute as fast as possible.
- C) The card must be fully tested prior to shipping.
- D) The tests must be executable by a low skilled operator without difficulty. This is to assure that product verification personnel do not require extensive experience and/or training to run the diagnostics efficiently.

The most important concern listed above is C). In other words, fault coverage is given priority over execution performance and "user-friendliness". The bottom line here is "make sure that if a defect exists, that it is detected".



Not surprisingly, the field diagnostician's concerns and the product verification personnel's concerns are quite similar. However, with respect to the field, the diagnostics may be executed on an on-line machine, which could have an undetected fault that affects the diagnostics (for simplicity, it is assumed that the machine running the diagnostics in product verification is functioning properly).

1.3.3 Defect Analysis's Diagnostic Concerns

The major concerns of the defect analysis personnel are as follows:

- A) Determine which component is defective.
- B) The test set should use an efficient coverage scheme (i.e. the tests should be able to home in on the faulty component efficiently).
- C) An automatic fault isolation capability would be beneficial, but the need for an interactive environment would still exist (an interactive environment would allow the user to run specific tests at will).
- D) Diagnostics should be in a top-down format since the user may wish to test specific "submodules" before isolating the fault to the component level (for example).

Basically, the above discussion implies that defect analysis would like a diagnostic which can isolate a fault to the component level in a proficient manner, since the duration of execution is an issue here, yet give the user ample flexibility to run tests as he/she desires.

1.4 METRICS TO BE USED

Metrics will be maintained on the software's performance. The software metrics will consist of bug status, in addition to bug related issues. That is to say, not only will software bugs be monitored, but the way in which the bug was made visible and the correction for that bug will also be recorded. Initially, all metrics will be recorded in a notebook by the diagnostic writer; however, before the diagnostic is released, all metrics will be included into the documentation concerning the AFIDAD.

·

PRODUCT ENVIRONMENT

SECTION

2.1 HARDWARE REQUIREMENTS

The hardware required to run the AFIDAD consists of a CIO AFI Device Adapter (Product Number: 27114A and 27114B), as well as a series 800 computer system with all the necessary equipment to bring the operating system (HP-UX) up and running.

Moreover, in order to <u>fully</u> test the AFI Device Adapter, a AFI loopback hood (part # 27114-60002) is required to be connected to the AFI. This will allow the frontplane drivers/receivers and external interrupt circuitry to be tested.

2.2 SOFTWARE REQUIREMENTS

In order to run the AFIDAD, the SHERLOCK diagnostic system must be present, along with the portability interface routines. Also, the diagnostic must guarantee that regardless of what condition the AFI Device Adapter (DA) to be tested is in, the system will not "crash" by running the diagnostic on it.

Other software that must be present in the system to execute the AFIDAD is the following: the AFI Device Access Routines (DAR) which act as an interface between the diagnostics and the *driver*, and the AFI Device Adapter Manager (DAM) which is the *driver* to the AFI DA. Note that the DAR is *technically* part of the portability interface, as far as the SHERLOCK module structure is concerned.

NOTE

The reason for implementing the DAR--instead of having a direct interface between the AFIDAD and the DAM--is to keep the AFIDAD independent of the operating system environment. For example, there are likely to be differences in how an operating system's services are accessed, which are masked to the diagnostic by implementing a DAR. Therefore, in order to use this diagnostic with a different operating system, only the DAR needs to be replaced. This tends to be a much "cleaner" implementation overall.

2.2.1 Software Compatibility

In order to test previous revisions of the AFI device adapter the AFIDAD must be able to provide diagnostics for various revisions of the hardware and various versions of the DAM. The following table shows these combinations.

DAM	Hardware
NEW	NEW
NEW	OLD
OLD	NEW
OLD	OLD

NOTE	

The new hardware is revision 2 of the AFI DA. The old hardware is any previous revision. The new DAM is the one written for the revision 2 AFI DA. The old DAM is any previous version.

The AFIDAD will have the ability to discern the DA revision and DAM version that is available. As a result the AFIDAD will be able to provide complete testing of all combinations, except one.

The old AFI DAM does not provide access to all of the features available in the revision 2 AFI DA. The AFIDAD will be unable to test these features which will reduce the effectiveness of the diagnostic for that combination of AFIDAD, DAM and DA.

2.3 DAD/DAR REQUIREMENTS

The diagnostic can only execute features of the DA that the DAR gives it access to and ,of course, the DAR can only execute features of the DA that the DAM gives its access to. In order for this diagnostic to be an effective tool, the DAR must support the functions listed below. Recall that the DAR relays all AFIDAD requests to the DAM, therefore the DAM must also support the functions illustrated below.

	1	V	0	•	T	I	

The DAR should always open the DAM in diagnostic mode.

The DAR/DAM must also ensure that no matter what the state of the device adapter is (functional, partially functional, or dead), the AFIDAD will regain system control within the timeout period (plus time for overhead operations) and that status information will also be returned to the AFIDAD after a command is issued to the DAR/DAM.



All DAM commands are issued by way of the DAR, which is invoked by the AFIDAD via the " afi_dar " program services routine. When this routine is called by the diagnostic, one of the parameters passed will specify the DAR function which is requested. When appropriate, other parameters will specify the particular address of a register to be read or written. For example, if a register write is desired, the function_code parameter is used to specify the register write request, while the reg_number parameter

is used to specify the particular register to be written. Additional information concerning the respective parameters is given below.

Also note that the DAR is responsible for making sure the correct DAM is allocated. This check should be done at the onset of every DAR invocation.

2.3.1 Read Device Adapter Register

The afi_read_reg function will be used by the AFIDAD in order to read an arbitrary register on the device adapter. Along with specifying this function to the DAR when " afi_dar " is called, the AFIDAD will also specify the register number of the desired register (within the DA's register space) whose contents are to be read in. The particular register number will be passed as a parameter to the DAR from the AFIDAD. After the respective register has been read by the DAM, the register's contents should be returned to the diagnostic. The diagnostic should have the capability to read <u>any one</u> of the registers implemented on the device adapter.

2.3.2 Write Device Adapter Register

The afi_write_reg function will be used by the AFIDAD in order to write data to an arbitrary register on the device adapter. Along with specifying this function to the DAR when " afi_dar " is called, the AFIDAD will also give the offset register (within the DA's register space) of the desired register whose contents are to be over-written. The particular offset address will be passed as a parameter to the DAR from the AFIDAD. The diagnostic should have the capability to write to <u>any one</u> of the registers implemented on the device adapter.

2.3.3 Reset Device Adapter

The *afi_reset* function must be able to put the AFI Device Adapter into a know state (power-on state) and clear any residual error conditions. The DAM should also be placed into a know state. Once the above is completed, the DAM should initialize the DA's registers with the necessary data to configure the card and bring it to an operational state.

2.3.4 Read Data

The afi_read function will be used by the AFIDAD when it desires data to be input from the AFI. In other words, whenever this function code is specified, the DAM should take the appropriate actions necessary to input data from the AFI.

When the DAM/DAR is requested to pass the incoming AFI data to the AFIDAD, it should continue to do so until one of the following conditions is met: 1) the specified number of bytes have been transferred, or 2) the device transfers all pertinent data.

2.3.5 Write Data

The afi_write function will be used by the AFIDAD when it desires data to be output to the AFI. In other words, whenever this function code is specified, the DAM should take the appropriate actions necessary to output data to the AFI.

2-3

2.3.6 Setup for Interrupt

The afi_setup_int function will be used to setup for and handle any pending interrupts from the AFI DA. An afi_check_int call must be made to fetch the status of any interrupt that happened after the setup call.

2.3.7 Check for Interrupt

The afi_check_int function will return a count of the number of interrupts that went off after the &feafi_setup_int call. A return value of zero indicates no interrupts, a value of indicates one interrupt, and so on.

2.3.8 CIO Poll status

The afi_poll function will be used to determine the CIO poll status of the AFI DA.

2.3.9 Get CIO Slot

The afi_getslot function will return the CIO slot number for the AFI DA. This slot number is used to obtain the correct poll status bit.

2.3.10 Identify DAM

The afi_identify function is used to determine the revision of the DAM. The value returned indicates a DAM available for a revision A AFI DA or a revision B AFI DA.

2.4 DIAGNOSTIC-DAR INTERFACE

As mentioned before, the AFIDAD cannot interact directly with the DAM, but rather must communicate with the DAM via the DAR, which is basically a buffer to isolate the diagnostic from the operating system. As such, all information that is necessary for the DAM/DAR to perform the function requested by the diagnostic, must be passed by the diagnostic as parameters. The parameters that will be passed to the routine " afi_dar " when it is called by the AFIDAD (as well as the reasons for passing them) are as follows:

DGLOBAL :	Contains	diagnostic	system	internal	information.
-----------	----------	------------	--------	----------	--------------

FUNCTION CODE : Function code for the desired operation.

- REG_NUMBER : Specifies a register number to be used with the respective AFI_READ_REG or AFI_WRITE_REG function code within the FUNCTION_CODE parameter. An entry corresponding to a function code which does not make use of this parameter (such as AFI_RESET_CARD) will have a value of zero.
- REG_VALUE : Specifies the address where the result of an AFI_READ_REG will be placed or where the value to be written in an AFI_WRITE_REG may be fetched.

* HP Confidential -- AFI Diagnostic ERS *

TIMEOUT : Time (in seconds) to wait for the respective operation to complete.

BUFF

: ID of buffer which contains the data to be written to or read from the device adapter.

BUFF LEN : ID of a short integer that contains the transaction count (in bytes) for the corresponding AFI READ or AFI WRITE function code within the FUNCTION CODE parameter. Only AFI READ and AFI WRITE function codes will have an entry within this parameter. If an entry corresponding to a AFI READ function code has a value of zero, an error condition exists, and the DAR should return the appropriate status value to the AFIDAD without further processing. Similarly, if an entry corresponding to a AFI WRITE function code has a value of zero, an error condition exists, and the DAR should return the appropriate status value to the AFIDAD without further processing of the command.

STATUS : Returns the success or failure of the operation. If the operation did in fact fail, an identifier describing the failure is returned.

The DAR function declaration should appear as follows:

struct diag sys type	<pre>#dglobal;</pre>
unsigned int	function code;
unsigned int	reg_number;
unsigned int	*reg value;
unsigned int	timeout;
struct buffer id type	<pre>#buff;</pre>
unsigned int	buff len;
short	*status;

CAUTION

This description of the parameters passed to the DAR is preliminary, and may be modified before the actual coding of the diagnostic commences.

2.5 DAR STATUS RETURNS

The following is a <u>preliminary</u> list of the status conditions that should be reported by the DAR. Note that the subset was extracted from the list maintained by SHERLOCK and only includes those status conditions which are most beneficial to the diagnostic. If the DAR is able to recognize additional status conditions

> * HP Confidential -- AFI Diagnostic ERS * 2-5

that may or may not be maintained by SHERLOCK, the added functionality will be included in the AFIDAD.

- # DIODEV TIMEOUT
- DIODEV FAIL *
- # DIODEV INTERNAL
- # DIODEVUNIMP
- **#** DIODEV BAD FUNC
- * DIODEV BAD DEV
- DIODEV BAD BUFF #
- *
- * DIODEV_DEST_FUNC

- /* DAR timed out */
- /* I/O failed due to device problems */
- /* DAR internal HW or SW error */
- /* specified function is not implemented */
- /* Function code is invalid */
- /* LDEV not allocated by "sel_sec_dev" */
- /* Pointer to an I/O buffer is NULL */
- DIODEV BUFF TOO SMALL /* I/O buffer is too small for function */
 - /* Function requires destructive mode, but not granted #/

EXTERNAL SPECIFICATIONS

SECTION 3

The AFI Device Adapter Diagnostic is part of the SHERLOCK diagnostic package. It is designed to provide its user with a means of determining if the specified AFI Device Adapter (DA) and its related hardware are operating properly, and if not, which FRU should be replaced. There are a variety of tests which the user can run to determine the source of a problem. Some of the tests require writing and reading data to and from (respectively) the DA after a loopback hood has been placed on the DA's frontplane.

3.1 USER ENVIRONMENTS

The AFIDAD can be run in in a one of the following two modes which are described below.

- Destructive Mode Indicates that the program may run any test it desires on the selected device. This mode is required for tests that have the potential for corrupting data on the device being tested (i.e. Destructive tests). There are virtually no restrictions on tests run in this mode; therefore, this mode is handled with extreme care by the diagnostic program. An example of a test that would require this mode in order to run would be one that reformats the media on a system disc, thus destroying all of the data on it.
- Non-Destructive Mode Indicates that the diagnostic program cannot run any tests on the selected device that are considered to be potentially destructive or disruptive in nature.

3.2 DIAGNOSTIC SYSTEM COMPOSITION

The SHERLOCK Diagnostic system divides diagnostic programs into three groups. These groups are as follows:

- Diagnostics and Verifiers Diagnostics are programs that are capable of determining which FRUs are malfunctioning in a device. Verifiers, on the other hand, cannot isolate defective FRUs, but can verify for the user which functions of the device are operating properly.
- Utilities and Tools These programs provide a means for obtaining system information or performing specific I/O operations.
- System Exercisers System exercisers are intended to provide a means of using (loading) a particular part of the system. These programs provide a way of using system resources under stress conditions that equal or exceed those expected under maximum load.

NOTE

The AFIDAD is classified as a diagnostic program, and every effort will be made to keep this diagnostic independent of the operating system.

3.3 USER INTERFACE

The AFIDAD can be accessed by users via the Diagnostic User Interface (DUI) provided by SHERLOCK. It is initiated by using the *run afidad* command. For further information on the *run* command provided by the DUI, please refer to the SHERLOCK DUI ES. All parameters available in the *run* command are acceptable as parameters when executing this diagnostic.

NOTE

All of the sections in this diagnostic can be executed from any terminal, even if a specific test requires the user to have the capability to run destructive tests. This implementation therefore allows the diagnostic to be run from a remote terminal; however, there may be system limitations that would not allow the use of a remote terminal.

When the AFIDAD is invoked, the following header and welcome messages will be displayed:

*********	* * * * * * * * * * * * * * * * * * * *	*******
****		*****
****	AFIDAD : AFI Device Adapter Diagnostic	*****
****		*****
***	(c) Copyright Hewlett-Packard Company 1987	*****
****	All Rights Reserved	*****
****		*****
****	Version V.UU.FF	*****
***		*****
***	******	*******

Welcome, Today is day, date time

After the header and welcome messages are displayed via a program services call, the diagnostic will call another program services routine in order to obtain access to the device that was selected for testing (in addition to setting up the sections and steps to be run).

This routine will exit with its status parameter (passed by reference) being any one of three possible values. The first of which is *successful*. This indicates that all sections and steps have been validated and that the system granted access to the device.

The second possible value is dssd_device_in_use. If this value is returned, it indicates that the system did not grant access to the device. If this happens, the following error message will be issued by the diagnostic:

***	ERROR	 AFI DEVICE ADAPTER ALREADY IN USE BY THE
***		DIAGNOSTIC SYSTEM.

***		Someone has already gained exclusive rights to the
***		AFI Device Adapter that you requested, and it is illegal
***		to have two copies of the AFIDAD diagnosing the same AFI
***		Device Adapter simultaneously.

The diagnostic will terminate execution after outputting this error message.

The third possible status value is *dssd_internal_error*. When this value is active upon exiting the subroutine, it indicates that an error such as no device adapter at the specified LDEV was found. SHERLOCK itself will output the error message for this situation, not the AFIDAD. The diagnostic will terminate upon regaining control.

If all went well up to this point, the sections and steps specified by the user will be executed and the results displayed. If the user did not specify any sections/steps to be run, the default sections and steps will be executed (Sections 3 and 6). If at any time, the number of errors generated exceeds the limit specified by the user in the errount parameter (of the DUI *run* command), the following message will be output:

*** WARNING -- The maximum specified number of error occurrences has
*** been exceeded.

The diagnostic will then terminate its execution. If the errpause parameter of the *run* command was assigned to "on", then the diagnostic will stop after each error is generated and ask the user if the testing should continue. The prompt that will be displayed is as follows:

Do you wish to continue? (Y/N) [Y] :

If the response is "Y", then the testing will resume (if possible), and if the response is "N", the diagnostic will terminate its execution. If the sections and steps specified by the user were executed the number of times specified in the loop parameter of the *run* command without the number of errors exceeding the erround value, the diagnostic will terminate normally.

At any time that the diagnostic is prompting the user for information, the user may enter "exit" to terminate its execution, or enter "suspend" to temporarily suspend its execution. Either the entire word or any number of characters which uniquely identify the respective language localized command may be entered. Moreover, the letters entered may be in any combination of upper and lower case characters. If the user exits in this fashion, the following message is displayed:

... Exiting AFIDAD per your request.

If the user temporarily suspends execution in this manner, the message that will be displayed is as follows:

... AFIDAD suspended per your request.

The user can then perform tasks through the Diagnostic User Interface (DUI) and subsequently resume execution of the AFIDAD, or he/she can abort the AFIDAD entirely.

3.4 AUTO-DIAGNOSTICS

Auto-diagnostics will no longer be supported under the converged Sherlock.

3.5 FUNCTIONAL DESCRIPTION OF SECTIONS

The remainder of this section discusses each section and step in detail. As a quick reference, the following table was included to list all of the sections and steps available for use in the AFIDAD.

NOTE

The reader should refer to Section 4, Help Messages, of this document for an in-depth description of the various device adapter registers accessible by this diagnostic software. This section (External Specifications) only gives a general description of a particular register (when appropriate) when illustrating the functionality of the respective section/step.

Section No.	Diagnostic Function
=======================================	
1	More Help
2	Reset
3	Identify
4	Hardware Test
5 .	Hardware Test with loopback
6	Status
7	Register Level Input/Output Transactions

CAUTION

The "Possible Error/Warning Messages" listed in the following sections are likely to change somewhat as the development of this diagnostic progresses. This is mainly because it is not known at this time exactly what conclusions can be drawn from a given error condition; however, this information will become available throughout the development phase, and will be used to update the message set so as to provide the user with the most precise messages possible.

3.5.1 Section 1 -- More Help

Minimum Mode Required : Normal Terminal Used for Execution : Any In Default Set? : No

More Help is is not longer supported under the converged Sherlock. The following message will be displayed if More Help is run:

Possible Output Messages:

Section 1 -- More Help

This Section is no longer supported. For additional information on a given section, use the sections option of the help command from the DUI prompt. For example, if you want more information about section 7, type the following:

DUI> help afidad sections 7

End of Section 1 -- More Help

AFI DAR Operations Used:

None

External Specifications

3.5.2 Section 2 -- Reset

Minimum Mode Required	:	Destructive
Terminal Used for Execution	:	Any
In Default Set?	:	No

This section informs the Device Adapter Manager (DAM)--via the Device Access Routines--to reset the AFI Device Adapter (DA) and DAM to its power-on state. The DAM will then transfer all pertinent data onto the DA that the DA needs to operate properly.

Possible Output Messages:

Section 2 -- Reset

NO ERRORS DETECTED while resetting the device adapter.

End of Section 2 -- Reset

Possible Error/Warning Messages:

*** ERROR -- An error occurred during reset. (AFIDADERR 202)
*** afi dar status returned = !

AFI DAR Operations Used:

afi_write_reg

3.5.3 Section 3 -- Identify

Minimum Mode Required : Destructive Terminal Used for Execution : Any In Default Set? : Yes

Identify displays the AFI product number and hardware revision code.

Possible Output Messages:

Section 3 -- Identify

CIO card ID byte = 32 Hardware Revcode = ? DAM available for hardware revision ?

```
End of Section 3 -- Identify
```

NOTE

The Hardware Version field may vary in time, therefore cannot be explicitly specified within this document.

Possible Error/Warning Messages:

*** ERROR -- CIO card ID byte = ! (UNKNOWN PRODUCT)
*** Expecting 32 (HP27114)

AFI DAR Operations Used:

afi_read_reg

External Specifications

3.5.4 Section 4 -- Hardware Test

Minimum Mode Required: DestructiveTerminal Used for Execution: AnyIn Default Set?: No

This section is a subset of section 5. Except for the frontplane handshake circuit and the interrupt circuit, all other circuits are tested. The diagnostic performs the following tests:

- INTERNAL REGISTER TEST identical register values and possible stuck bits are looked for.
- REVISION TEST the revision and ID register is verified.
- READ STATUS the status is read and verified.
- RESET CARD the AFI DA is reset and verified.
- POLL RESPONSE TEST the polling circuit is verified.
- FIFO TEST the FIFO and related circuits are tested by filling and emptying the FIFO and checking the status at various times.
- TRANSFER COUNTER TEST the transfer counter is tested.
- READ/WRITE TEST Various patterns (0, all 1s, 5s, As, walking 1) are written out to and read back from the FIFO.

Possible Output Messages:

Section 4 -- Hardware test

CIO card ID byte = 32 Hardware Revcode = ? DAM available for hardware revision ?

No hardware errors found.

End of Section 4 -- Hardware test

Possible Error/Warning Messages:

***** ERROR -- HARDWARE ERROR DETECTED (AFIDADERR !)**

*** ERROR -- RESOURCE ALLOCATION ERROR IN SYSTEM (AFIDADERR ?) *** TEST IS TERMINATED

AFI DAR Operations Used

- -

afi_read_reg afi_write_reg afi_read afi_write afi_poll afi_getslot **External Specifications**

3.5.5 Section 5 -- Hardware Test with Testhood

Minimum Mode Required : Destructive Terminal Used for Execution : Any In Default Set? : No

Executing this section will perform a thorough hardware test on the AFI DA. The glue logic and backplane transceivers will also be tested. The only components that will be untested after this section is executed are the glue logic that is inaccessible to the AFIDAD. In order to run this section, a loopback hood must first be connected to the respective device adapter. After the user confirms that the loopback hood has been connected, the diagnostic performs the following specific tests:

- INTERNAL REGISTER TEST identical register values and possible stuck bits are looked for.
- REVISION TEST the revision and ID register is verified.
- INTERRUPT TEST the interrupt circuit is tested and verified. Tests are made to verify that the system acknowledges the interrupts.
- READ STATUS the status is read and verified.
- RESET CARD the AFI DA is reset and verified.
- POLL RESPONSE TEST the polling circuit is verified.
- FIFO TEST the FIFO and related circuits are tested by filling and emptying the FIFO and checking the status a various times.
- TRANSFER COUNTER TEST the transfer counter is tested.
- READ/WRITE TEST Various patterns (0, all 1s, 5s, As, walking 1) are written out to and read back from the FIFO.
- LOOPBACK TEST the data path to and from the frontplane is tested using various patterns (0, all 1s, 5s, As, walking 1).

Possible Output Messages:

Section 5 -- Hardware test with testhood

CIO card ID byte = 32 Hardware Revcode = ? DAM available for hardware revision ?

No hardware errors found.

End of Section 5 -- Hardware test with testhood

* IIP Confidential -- AFI Diagnostic ERS * 3-10

Possible Error/Warning Messages:

******* ERROR -- HARDWARE ERROR DETECTED (AFIDADERR !)

*** ERROR -- HARDWARE ERROR DETECTED (AFIDADERR !) *** VERIFY TESTHOOD IS IN PLACE

*** ERROR -- RESOURCE ALLOCATION ERROR IN SYSTEM (AFIDADERR !) *** TEST IS TERMINATED

*** WARNING -- TEST HOOD APPEARS TO BE MISSING. PLEASE CHECK.

Continue (yes/no) <yes> ?

AFI DAR Operations Used

afi_read_reg afi_write_reg afi_read afi_write afi_poll afi_getslot **External Specifications**

3.5.6 Section 6 -- Status

Minimum Mode Required : Destructive Terminal Used for Execution : Any In Default Set? : Yes

Running this section will result in status information of the interface with the peripheral to be displayed in as descriptive sentences rather than bit positions within the status register.

Possible Output Messages:

Section 6 -- Status

ARQ interrupt is pending/clear.

ARQ interrupt is enabled/disabled.

CEND is asserted/de-asserted.

FIFO has room for data/is full.

FIFO has data in it/no data in it.

PFLG is asserted/de-asserted.

PCTL is asserted/de-asserted.

Transfer counter equals zero/is not equal to zero.

Transfer counter value is ?

ATTN is reset/set.

DEND is reset/set.

FIFO contains aproximately 0 - 8 words 9 - 31 words 32 - 55 words 56 - 64 words

State of status lines STSO through STS5 (in that order): Asserted De-asserted

End of Section 6 -- Status

Possible Error/Warning Messages:

None specified by the diagnostic.

* HP Confidential -- AFI Diagnostic ERS * 3-12

AFI DAR Operations Used:

afi_read_reg

3.5.7 Section 7 -- Register Level Input/Output Transactions

Minimum Mode Required : Destructive Terminal Used for Execution : Any In Default Set? : No

WARNING

The user of this section must exercise EXTREME CAUTION when sending information over the AFI.

This section can be executed by the diagnostician when he/she wishes to do "peeks" and/or "pokes" to the device adapter. That is to say, this section allows the user to read/write registers in the device adapter's address space.

WARNING

In order for this section to be of any value, the device adapter being diagnosed should not be accessed by any process other than the AFIDAD. This is to ensure that any configuration modifications made by the diagnostic are pending for succeeding transactions.

Possible Output Messages:

Section 7 -- Register Level Input/Output Transactions

Enter register selection, read/write and output data (if applicable) Use register 2 for poll function. Last entered value for same question is repeated if carriage return is entered.

Register number (0, 1, 2 [POLL], 3, 7, 9, 10, 11, exit) =

Input 0 for read, 1 for write:

End of Section 7 -- Register Level Input/Output Transactions

NOTE

All commands/data can be entered in upper and/or lower case characters.

Possible Error/Warning Messages:

******* Value ! is illegal

******* Illegal Register Address

******* Superuser priviledge required for poll response testing

******* This value (!) is illegal for this register (0)

******* This value (!) is illegal for this register (1 or 7)

AFI DAR Operations Used:

afi_write_reg afi_read_reg The purpose of this section is to describe the help messages that will be included in the message catalog. Some of these messages are displayed if the user issues the *help afidad* command at the DUI prompt, or the help afidad sections command.

4.1 MESSAGE 20

Purpose:

This message contains a general description of the diagnostic program, including a general overview of what the purpose of the program is, what it is capable of doing, and what capabilities the user must possess in order to run it.

Text:

AFIDAD is a diagnostic program used to diagnose the condition of the AFI (27114A/B) 16-bit parallel interface device adapter. The user can verify the revision and data code of the DA and its device adapter manager (DAM, I/O driver). The card can be subjected to two tests, a limited hardware test or an expanded test with loopback. These tests cover all of the testable circuitry on the DA. This diagnostic does not attempt to diagnose the user's peripheral connected to the AFI DA.

4.2 MESSAGE 10001

Purpose:

This message is supposed to contain a list of commands available in the diagnostic, however since this diagnostic is not interactive, it does not contain any commands. (An example of an interactive diagnostic is TERMDSM. When this diagnostic is invoked, it displays a user prompt and it executes the commands entered. The AFIDAD only executes the sections and steps which SHERLOCK has decreed appropriate, and is therefore not interactive.)



There are certain sections/steps within the AFIDAD that request information from the user, that is to say, they are interactive. However, an interactive section/step does not an interactive diagnostic make.

Text:

Help Messages

AFIDAD is not an interactive diagnostic. It contains no commands.

4.3 MESSAGE 10002

Purpose:

This message contains a list of the parameters in the DUI run command that are meaningful to the program.

Text:

All parameters in the RUN command are accepted by AFIDAD.

4.4 MESSAGE 30

Purpose:

This message contains a list (and short descriptions) of the sections and steps that are available in the AFIDAD. Also included is the user capability required to run the respective sections.

Text:

```
Section 1 - More Help
    [Normal Mode, Not in Default Set]
    Gives futher information about each of the sections.
Section 2 - Reset
    [Destructive Mode, Not in Default Set]
    Resets the AFI Device Adapter (DA)
Section 3 - Identify
    [Normal Mode, In Default Set]
    Information is returned such as the various date codes.
Section 4 - Test
    [Destructive Mode, Not in default Set]
    Performs a test on the AFI DA, including limited data loopback via
    the CIO backplane.
Section 5 - Test with test hood
    [Destructive Mode, Not in Default Set]
    Performs a test on the AFI DA, including data loopback via the CIO
    backplane, FIFO and test hood. This is the most complete test for
    the AFI DA. This test requires the test hood to be installed.
Section 6 - Status
    [Normal Mode, Not in Default Set]
    Obtains status from the AFI DA and displays it in human readable form.
* HP Confidential -- PB AFI Diagnostic ERS *
4-2
```

```
Section 7 - Control
[Destructive Mode, Not in Default Set]
Send control to AFI state machine then obtain status from AFI DA and
display it.
```

4.5 MESSAGES FOR SECTION HELP

The following help messages are displayed within the DUI by typing help afidad sections N where "N" is a section number in the range 1 - 7. These messages give detailed help for each of the sections.

4.5.1 Set 6, message 1

Text:

Section 1 - More Help. This section is no longer supported. For more information about a section, type "help afidad sections N" where "N" is a section number in the range 1 - 7.

4.5.2 Set 6, message 2

Text:

Section 2 - Reset. This allows the user to reset the AFI DA. This section is destructive by its nature. It is possible that data that is currently on the AFI DA will be lost. This test should only be run when absolutely necessary.

4.5.3 Set 6, message 3

Text:

Section 3 - Identify. This section is used to determine that the device under test is indeed a valid AFI DA and to display the various operating parameters of the interface. This section can be run at any time and is a NORMAL mode test. The following information is displayed in this section:

Product Number - This is the product number of the AFI DA that is currently being diagnosed.

Hardware Revcode - This is the Revision of the physical PCBA.

4.5.4 Set 6, message 4

Text:

Section 4 - Test. The test can be run regardless what type of device or test hood is connected to its front-plane connector. This test includes a data loopback via its on board FIFO. This section is destructive and should only be done when necessary.

4.5.5 Set 6, message 5

Text:

Section 5 - Test with test hood. This includes a loopback test via the FIFO and the external test hood. This test is destructive and should be done only when necessary. This is the most complete test for the AFI DA.

4.5.6 Set 6, message 6

Text:

Section 6 - Status. This section gets status from the AFI DA and then decodes this status. The following status info is obtained:

The ARQ (Attention Request) interrupt ARQ enable flipflop Readiness of output and input of the FIFO Condition of the transfer counter Logic state of handshake lines IR and OR Logic state of handshake lines PCTL and PFLG Logic state of status lines STSO, STS1, STS2, STS3, STS4, STS5

The above information is read from register 1 and register 6.

4.5.7 Set 6, message 7

Text:

Section 7 - Control. This section provides direct access to all AFI registers. The section loops until the word 'exit' is entered at the register input prompt. A carriage return $\langle CR \rangle$ at this prompt repeats the input provided from the previous register input prompt. If the register requested is a read/write register a prompt requesting the type of operation is displayed. A carriage return $\langle CR \rangle$ at this prompt repeats the input provided from the previous operation prompt. If the register requested is a read only register, a read operation is done

automatically. Poll response of the AFI card is also available as register 2.

This is most useful as a tool to debug the DA and its interface with a user's equipment. Except for the register being accessed, the device adapter is not affected in any other way. Please refer to user's manual for possible uses of these directly accessible registers. This section is destructive and should be run only when necessary. The following is a simple example of using the register writes and reads:

WRITE:

- Write to register 7 the value 16 (binary 00010000) to define output mode, enable poll response, clear FIFO, enable peripheral handshake and use the rising edge of PFLG as the triggering edge. To preserve what was in the FIFO, this previous step should be skipped.
- Write to register 7 the value 0 (binary 0000000) to enable the FIFO and maintain other definitions.
- Determine if the FIFO is full by reading the poll response. Register 2 is reserved for this purpose. If poll response is negative, the FIFO is full and any further write to register 0 will be lost.

Write to register 0 the data unit when poll response is positive.

READ:

- Write to register 7 the value 80 (binary 01010000) to define input mode, enable poll response, clear FIFO, enable peripheral handshake and use the rising edge of PFLG as the triggering edge. To preserve what was in the FIFO, this previous step should be skipped.
- Write to register 7 the value 64 (binary 01000000) to enable FIFO and maintain other definitions.
- Determine if data is put in the FIFO by the peripheral by reading the poll response. Register 2 reading is reserved for this purpose. Note that there is no actual register 2 on this device adapter. If poll response is negative, there is no valid data in the FIFO (reading from register 0 in this condition returns invalid data).

Read a valid data word from register 0.

NOTE:

Loopback data via the FIFO or testhood involves more steps than above. Please refer to the user's manual for a complete description.