Instruction Manual

Tektronix

DAS 92DM75A
R4000PC, R4000SC & R4000MC
Microprocessor Support

070-8750-00

Warning
The servicing instructions are for use by qualified personnel only. To avoid personal injury, do not perform any servicing unless you are qualified to do so. Refer to the Safety Summary prior to performing service.

Please check for change information at the rear of this manual.

First Edition: March 1993
Online Version: April 1997

Printed in the U.S.A.

Tektronix, Inc., P.O. Box 1000, Wilsonville, OR 97070–1000

WARRANTY

Tektronix warrants that this product will be free from defects in materials and workmanship for a period of one (1) year from the date of shipment. If any such product proves defective during this warranty period, Tektronix, at its option, either will repair the defective product without charge for parts and labor, or will provide a replacement in exchange for the defective product.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period and make suitable arrangements for the performance of service. Tektronix will provide such service at Customer's site without charge during the warranty period, if the service is performed within the normal on-site service area. Tektronix will provide on-site service outside the normal on-site service area only upon prior agreement and subject to payment of all travel expenses by Customer. When or where on-site service is not available, Customer shall be responsible for packaging and shipping the defective product to the service center designated by Tektronix, with shipping charges prepaid. Tektronix shall pay for the return of the product to Customer if the shipment is to a location within the country in which the Tektronix service center is located. Customer shall be responsible for paying all shipping charges, duties, taxes, and any other charges for products returned to any other locations.

This warranty shall not apply to any defect, failure or damage caused by improper use or improper or inadequate maintenance and care. Tektronix shall not be obligated to furnish service under this warranty a) to repair damage resulting from attempts by personnel other than Tektronix representatives to install, repair or service the product; b) to repair damage resulting from improper use or connection to incompatible equipment; or c) to service a product that has been modified or integrated with other products when the effect of such modification or integration increases the time or difficulty of servicing the product.

THIS WARRANTY IS GIVEN BY TEKTRONIX WITH RESPECT TO THIS PRODUCT IN LIEU OF ANY OTHER WARRANTIES, EXPRESSED OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX' RESPONSIBILITY TO REPAIR OR REPLACE DEFECTIVE PRODUCTS IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

WARRANTY

Tektronix warrants that this software product will conform to the specifications in the documentation provided with the product, when used properly in the specified operating environment, for a period of three (3) months. The warranty period begins on the date of shipment, except that if the program is installed by Tektronix, the warranty period begins on the date of installation or one month after the date of shipment, whichever is earlier. If this software product does not conform as warranted, Tektronix will provide remedial services as described in the documentation provided with the product. Tektronix does not warrant that the functions contained in this software product will meet Customer's requirements or that operation of the programs will be uninterrupted or error-free or that all errors will be corrected.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period and make suitable arrangements for such service in accordance with the instructions received from Tektronix. If Tektronix is unable, within a reasonable time after receipt of such notice, to provide remedial services, Customer may terminate the license for this software product and return this software product and any associated materials to Tektronix for credit or refund.

This warranty shall not apply to any software product that has been modified or altered by Customer. Tektronix shall not be obligated to furnish service under this warranty with respect to any software product a) that is used in an operating environment other than that specified or in a manner inconsistent with the User Manual and documentation or b) when the software product has been integrated with other software if the result of such integration increases the time or difficulty of analyzing or servicing the software product or the problems ascribed to the software product.

THIS WARRANTY IS GIVEN BY TEKTRONIX WITH RESPECT TO THE LISTED PRODUCT IN LIEU OF ANY OTHER WARRANTIES, EXPRESSED OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX' RESPONSIBILITY TO PROVIDE REMEDIAL SERVICE WHEN SPECIFIED, REPLACE DEFECTIVE MEDIA, OR REFUND CUSTOMER'S PAYMENT IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Table of Contents

List of Figures

List of Tables

Preface: GUIDE TO DAS 9200 DOCUMENTATION

The Digital Analysis System (DAS) 9200 documentation package provides the information necessary to install, operate, maintain, and service the DAS 9200. The DAS 9200 documentation consists of the following:

- a series of microprocessor-specific **microprocessor support instructions** that describe the various microprocessor support packages
- a **system user manual** that includes a beginning user's orientation, a discussion of DAS 9200 system-level operation, and reference information such as installation procedures, specifications, error messages, and a complete system glossary
- a series of **module user manuals** that describe each of the DAS 9200 acquisition, pattern generation, and optional I/O modules
- an **on-line documentation** package that includes "context-sensitive" technical notes
- a **programmatic command language user manual** that describes the set of programmatic commands available for remotely controlling the DAS 9200
- a series of **application software user manuals** that describe the various application software packages
- a **technician's reference manual** that helps a qualified technician isolate DAS 9200 problems to the individual module level and determine corrective action (including on-site removal and replacement of modules)
- a **verification and adjustment procedures manual** that allows a qualified technician to make necessary adjustments and verify specifications of the mainframe and modules
- a series of **workbooks** that teach concepts about DAS 9200 acquisition modules and pattern generation modules

GENERAL SAFETY SUMMARY/ MICROPROCESSOR SUPPORT

The general safety information in this summary is for operating and servicing personnel. Specific warnings and cautions can be found throughout the manual where they apply and may not appear in this summary. While using this product you may need to access parts of the mainframe system; if so, read the General Safety Summary in your system user manual for warnings and cautions related to operating the mainframe system.

TERMS IN THIS MANUAL

CAUTION

CAUTION statements identify conditions or practices that could result in damage to the equipment or other property.

WARNING

WARNING statements identify conditions or practices that could result in personal injury or loss of life.

TERMS AS MARKED ON EQUIPMENT

CAUTION indicates a hazard to property, including the equipment itself, and could cause minor personal injury.

WARNING indicates solely a personal injury hazard not immediately accessible as you read the marking.

DANGER indicates a personal injury hazard immediately accessible as you read the marking.

SYMBOLS AS MARKED ON EQUIPMENT

DANGER—High voltage.

Protective ground (earth) terminal.

ATTENTION—REFER TO MANUAL.

USE CARE WITH COVERS REMOVED

To avoid personal injury, remove jewelry such as rings, watches, and other metallic objects before removing the cover. Do not touch exposed connections and components within the product while the power cord is connected.

REMOVE FROM OPERATION

If you have reason to believe that the instrument has suffered a component failure, do not operate the instrument until the cause of the failure has been determined and corrected.

USE THE PROPER FUSE

To avoid fire hazard use only a fuse of the correct type, voltage rating, and current rating.

REMOVE LOOSE OBJECTS

During disassembly or installation procedures, screws or other small objects may fall to the bottom of the mainframe. To avoid shorting out the power supply, do not power on the instrument until such objects have been removed.

DO NOT OPERATE IN EXPLOSIVE ATMOSPHERES

To avoid explosion, do not operate this product in an explosive atmosphere unless it has been specifically certified for such operation.

Section 1: OVERVIEW

The 92DM75A Microprocessor Support product displays data from systems based on the R4000PC, R4000SC, and R4000MC microprocessors. The 92DM75A product runs on a DAS 9200 system equipped with a 92A96 Data Acquisition Module.

This product consists of software on a floppy disk, a probe adapter, and this manual. The software includes setup files.

Included in the software files is a demonstration reference memory file. This file (called R4000_Demo) shows disassembled data; it is automatically installed on the DAS 9200 when you install the software. Most figures in Section 3 that show acquired data are taken from this demonstration reference memory. Directions for viewing this file are in Section 2.

BASIC INFORMATION

To use this product, you need to have the following:

- this manual
- other DAS 9200 mainframe and data acquisition module user manuals
- knowledge of your specific DAS 9200 configuration and its operation
- the *MIPS R4000 Microprocessor User's Manual* (1991)
- knowledge of your R4000PC, R4000SC, or R4000MC system

DAS 9200 System Software Compatibility

The R4000 disassembly software is compatible with DAS 9200 System Software Release 3, Version 1.3 or greater.

About This Manual

The organization of this manual is based on the sequence of steps necessary to use the disassembler. This manual provides detailed information on how to do the following:

- install software
- connect to your system under test
- setup support software and use it
- view acquired data
- maintain the probe adapter

Manual Conventions

The following conventions are used in this manual:

- the terms disassembler and disassembler software are used interchangeably in reference to the 92DM75A software that disassembles the bus cycles into instruction mnemonics and cycle types
- the terms system under test and SUT are used to refer to the microprocessor system under test
- references to the R4000 microprocessor apply to the R4000PC, R4000SC, and R4000MC microprocessors unless otherwise noted
- references to the 92A96 Data Acquisition Module include all versions of that module unless otherwise noted
- a signal that is active low has an asterisk (*) following the signal name, for example, ValidIn*
- signals that are latched have an _L following the signal name, for example, RdRdy*_L
- signals that are joined with other signals have a _D following the signal name which means the signals are derived, for example, Int*_D
- signals that are double probed have an equal sign (=) following the signal name, for example, ValidOut*=

Other Necessary Manuals

Before using these instructions, you should be familiar with the operation of a DAS 9200 with the data acquisition module you are using. For general instructions on the use of the DAS 9200 and a data acquisition module, refer to both the *DAS 9200 System User Manual* and the *92A96 Module User Manual.*

Refer to the *MIPS R4000 Microprocessor User's Manual* (1991) for information about the R4000 microprocessor.

DAS 9200 Configuration

To use the R4000 microprocessor support package, your DAS 9200 must be equipped with at least one 92A96 Data Acquisition Module with probe cables and standard probes.

The standard probe assemblies consist of four probe sets. Each probe set consists of one clock data probe with three 8-channel data probes each. The clock probe (a single channel), and each channel of the 8-channel probe, has one signal connection and one ground connection. Leadsets and KlipChips are not required.

Figure 1-1 shows an overview of the DAS 9200 connected to the system under test.

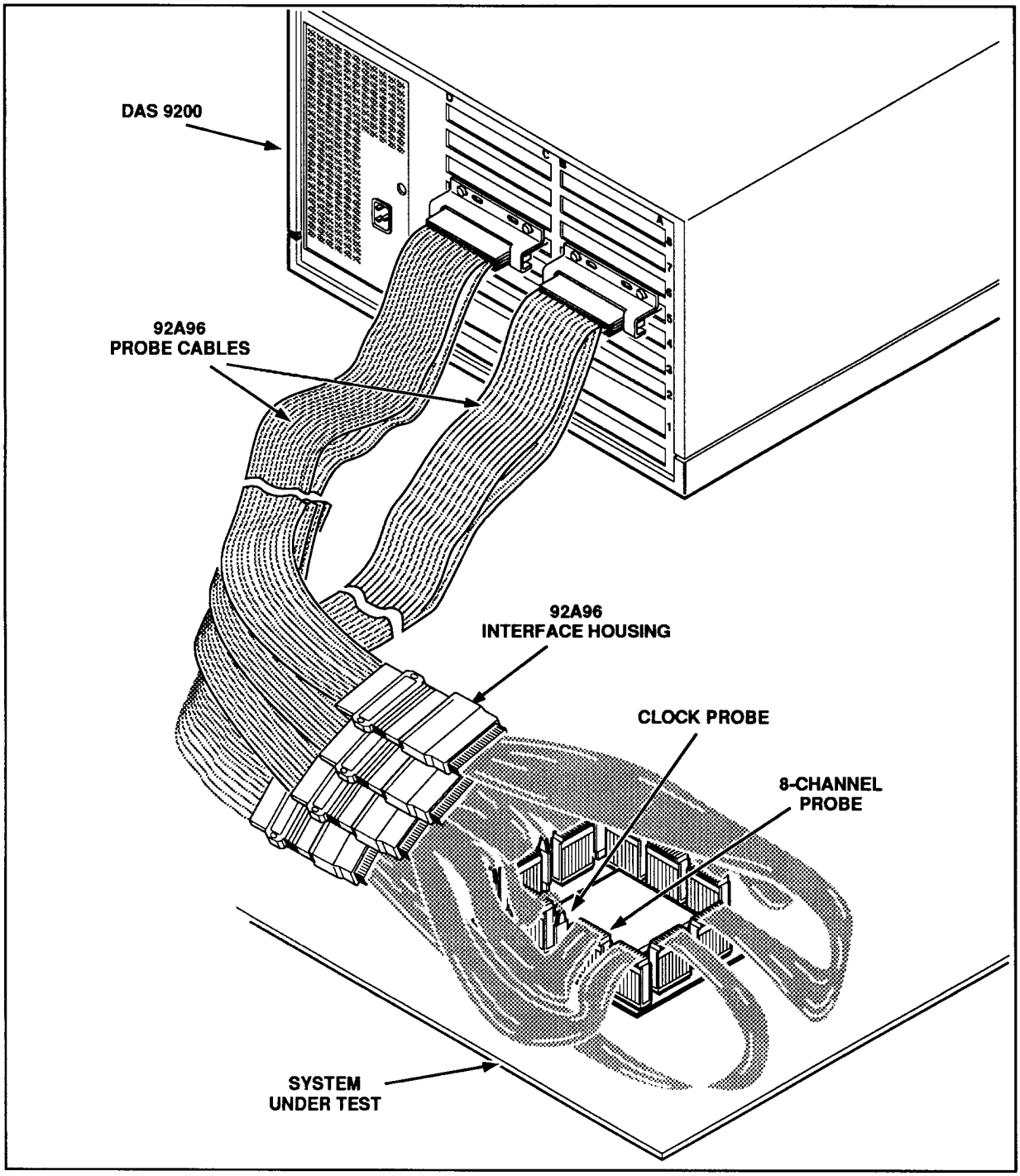


Figure 1-1. Overview of a DAS 9200 connected to a system under test.

R4000 SYSTEM REQUIREMENTS AND RESTRICTIONS

You should consider certain system requirements and restrictions of the R4000 microprocessor before using the disassembler. You should also consider all electrical, environmental, and mechanical specifications in Appendix C as they pertain to your system under test. The remainder of this section describes R4000 constraints.

System Clock Rate. The microprocessor support package supports the R4000 microprocessor running up to 50 MHz[1] .

Pins AA35 and AA39 (R4000SC). Your R4000SC system must not connect the AA35 and AA39 pins to ground or to a logic-low voltage; you can connect them to a logic-high voltage or leave them floating.

Secondary Cache (R4000SC and R4000MC only). The option 2S probe adapter does not probe the secondary cache of the R4000SC or R4000MC microprocessors. The probe adapter only probes the System Interface bus.

Translation Lookaside Buffer (TLB). The R4000 microprocessor contains a built-in TLB that performs virtual-to-physical address translation. Because of this, all addresses seen by the disassembler are physical addresses. This affects the way that symbol tables are defined for the Addr (Lo) group, and the reliability of the calculation of PC-relative addresses.

PC-with-Displacement Calculations. The disassembler displays PC-relative branch instructions with absolute addresses calculated from the current address of the program counter (PC). Since the disassembler sees the current address of the PC as a physical address, the calculated absolute branch address will also be a physical address. This means that the displayed branch address will not usually match the virtual address specified by the microprocessor software.

You should also be aware that incorrect physical branch addresses my be calculated and displayed when program execution crosses TLB page boundaries and the virtual-to-physical mapping changes.

---

[1] Specification at time of printing. Contact your DAS 9200 sales representative for current information on the fastest devices supported.

In cases where the current address of the PC plus the defined offset would normally cause a carry from the low-order 32 bits of the address into the high-order 4 bits, the carry will not occur.

However, if your R4000 system's assembler/compiler has defined all address symbols with a virtual address that is relative to the origin of your code, and your code is loaded into memory as one contiguous block, then you can build a symbol table using virtual addresses. To do this, you must select Relative to Base in the Mode field of the Symbol Editor menu. You must also enter the physical address offset of the origin in the Base field in the Symbol Editor menu.

Probe Adapter Clearance. Your R4000 system must have a minimum amount of clear space surrounding the R4000 microprocessor to accommodate the standard (R4000PC) or option 2S (R4000SC/R4000MC) probe adapter. Figures C-1 and C-2 in *Appendix C: Service Information* show these dimensions.

R4000 System and Probe Adapter Cooling. You must be sure to retain the original level of cooling for your R4000 system after you install the probe adapter. To maintain the required operating temperature, you may need to provide additional cooling for the microprocessor, which is surrounded by the probe adapter and probe connections.

Probe Loading. Any electrical connection to your system adds an additional ac and dc load. The 92A96 probes and 92DM75A probe adapter were carefully designed to add the minimum possible load to your system. This additional load may affect the operation of the R4000 in systems with extremely tight timing margins. Appendix C contains complete specifications on how the R4000 probe adapter affects your system.

Section 2: INSTALLATION AND CONNECTIONS

This section contains detailed descriptions of how to do the following:

- install the disassembler software
- view the demonstration reference memory
- set up the disassembler software
- connect the DAS 9200 to the system under test (SUT)
- configure the probe adapter

You must install the microprocessor support software prior to invoking and setting up the disassembler. It does not matter if you install the software before or after making the DAS 9200 connections and configuring the probe adapter.

INSTALLING SOFTWARE

The disassembler software sets up the DAS 9200 to acquire, disassemble, and display data from an R4000 system. To install the software, the application files on the 5.25 inch floppy disk must be copied to the DAS 9200 hard disk.

The DAS 9200 will not allow you to install the 92DM75A software if 92DM75 software is already installed on the system. The DAS 9200 will not overwrite 92DM75 files with 92DM75A files. Therefore, you will have to remove 92DM75 software prior to installing 92DM75A software.

However, the DAS 9200 will overwrite 92DM75A files if you reinstall 92DM75 software. You should remove 92DM75A software before reinstalling 92DM75 software.

To install the software, follow these steps:

1. Power on the DAS 9200 and press the Select Menu key.
2. Select the Disk Services menu in the Utilities column.
3. Press F6: MOVE TO UTILITY.
4. Select Install Application in the Operation field.
5. Press F8: EXECUTE OPERATION and follow the on-screen prompts.

If there is inadequate disk free space available on the hard disk, you must use the Remove Application or Delete File function of the Disk Services menu to free up enough disk space to install the software. The approximate space required to install the software is listed on the label of the floppy disk.

After the DAS 9200 successfully copies the application files from the floppy disk to the hard disk, the message **Application installation complete with no errors** appears on your screen. Remove the floppy disk and store it in a safe place in case you need to reinstall the software.

If you would like to see an example of R4000 bus activity with mnemonic disassembly, read the next discussion and procedure.

Viewing the Refmem File

A reference memory file is provided for you to familiarize yourself with the way the disassembler displays R4000 cycle types and instruction mnemonics. You can select the reference memory file to see how R4000 mnemonics are displayed without making any DAS 9200 connections to your system under test.

The R4000_Demo reference memory file is automatically installed when the disassembler software is installed on the hard disk. All the figures in Section 3 showing acquired data are from this file. The data you acquire from your R4000 system will be different.

To view the R4000_Demo Refmem, use the following procedure:

1. Press the Select Menu key to return to the Menu Selection overlay.
2. Move the cursor to the Refmem column and select the R4000_Demo file.
3. Move the cursor to the Display column and select Disasm.
4. Press F5: MOVE TO DISPLAY or the Return key to view the reference memory.

You can change the format of disassembled data from the Disassembly Format Definition overlay, which you can access through the Disassembly menu. Hardware disassembly is the default format. Examples of the data display formats are found under *Display Formats* in Section 4.

If there is not enough free space left on the hard disk, you can delete the R4000_Demo files (reference memory and symbol table) from the hard disk. They are presented strictly for viewing and are not needed to use the disassembler.

Setting Up Disassembler Software

The microprocessor support package supplies the disassembler software and setup files for the data acquisition module to use to acquire and display instruction mnemonics. Setup files are supplied for the Channel, Clock, and Trigger menus. Symbol files are supplied for displaying data and to use in the Trigger menu. A format file is also provided for the Timing menu when performing hardware analysis (described in Section 4).

Before selecting and setting up the disassembler, read the descriptions of *System Requirements and Restrictions* in Section 1.

If you have more than one data acquisition module in the DAS 9200, you should apply slot number labels to various parts of the DAS 9200 equipment before setting up the disassembler. Refer to the *Labels* discussion in this section for a description of where to apply the slot number labels. These slot numbers will help you identify which data acquisition module is connected to the probe adapter in a multimodule system.

When there are two or three 92A96 Modules in adjacent slots, they are automatically formed into a variable-width module by the system software at power up. If you need to use one 92A96 Module from a variable-width module, you must reconfigure the DAS 9200 accordingly prior to selecting software support in the 92A96 Configuration menu. Refer to the discussion of the System Configuration menu in the *DAS 9200 System User Manual* for details on how to reconfigure variable-width modules.

You can invoke the disassembler and its associated setup files from the module's Configuration menu. Move the cursor to the Software Support field and select R4000 Support. The support software automatically sets up the various module menus as soon as you select R4000 Support.

A detailed description of the data display setup follows. You can select the Channel, Clock, Trigger, and Symbol Editor menus to view default setups and files.

Channel Groups and Assignments

The disassembler relies on the presence of the signals and channel groups defined by the support software for the microprocessor. The channel groups for the R4000 microprocessor are Data(Hi), Addr(Lo), Control, Intr, Ack, A, Misc1, and Misc2. If you want to know which signal is in which group, refer to the channel assignment table in *Appendix C: Service Information*.

What You Can Change During Setup

You can change part of the module setup without affecting displayed data. You can change the trigger program in the Trigger menu and the display radix for any channel group in the Channel menu or State Format Definition overlay. (The A, Misc1, and Misc2 groups are not visible by default.) Channel assignments for all channel groups are located in Appendix C.

You cannot change the channel grouping or name for the Data(Hi), Addr(Lo), or Control groups, the threshold voltage, or polarity and expect the data display to function properly.

You can change the R4000 signal connections, the channel grouping, and group name for the channels not required for mnemonic disassembly. Refer to the discussions on *Signals Not Acquired* in Appendix B for a list of these signal connections.

Symbol Tables

You can use symbol tables to display channel group information symbolically in the State and Disassembly menus and to control triggering. The disassembler software contains symbol table files for the Control, Interrupt, and Acknowledge channel groups named R4000_Ctrl, R4000_Intr, and R4000_Ack. These symbol tables assign symbolic names to combinations of signal values within the Control, Interrupt, and Acknowledge channel groups.

Table 2-1 shows the name, bit pattern, and meaning for the symbols in the file R4000_Ctrl, the Control group symbol table.

Table 2-1
Control Group Symbol Table (R4000_Ctrl)

Symbol	Control Group Value Reset* ValidIn* ValidOut* SysCmd8 SysCmd7 SysCmd6 SysCmd5 SysCmd4 SysCmd3 SysCmd2 SysCmd1 SysCmd0	Meaning
Miscellaneous Cycles		
Reset	0 X X X X X X X X X X X X	Reset
Bus_Fault	1 0 0 X X X X X X X X X X	Bus fault
Bus_Idle	1 1 1 X X X X X X X X X X	Bus idle
Processor Commands		
Pr_RwWF	1 X 0 0 0 0 1 X X X X X †	Read with write forthcoming
Pr_RwWF_Coherent_Excl	1 X 0 0 0 0 1 0 1 X X X †	Coherent block, exclusive
Pr_RwWF_Coherent_Excl_Link	1 X 0 0 0 0 1 0 1 1 X X †	Link address retained
Pr_RwWF_Coherent_Excl_Link_4w	1 X 0 0 0 0 1 0 1 1 0 0	4 words
Pr_RwWF_Coherent_Excl_Link_8w	1 X 0 0 0 0 1 0 1 1 0 1	8 words
Pr_RwWF_Coherent_Excl_Link_16w	1 X 0 0 0 0 1 0 1 1 1 0	16 words
Pr_RwWF_Coherent_Excl_Link_32w	1 X 0 0 0 0 1 0 1 1 1 1	32 words
Pr_RwWF_Coherent_Excl_NoLink	1 X 0 0 0 0 1 0 1 0 X X †	Link address not retained
Pr_RwWF_Coherent_Excl_4w	1 X 0 0 0 0 1 0 1 X 0 0	4 words
Pr_RwWF_Coherent_Excl_8w	1 X 0 0 0 0 1 0 1 X 0 1	8 words
Pr_RwWF_Coherent_Excl_16w	1 X 0 0 0 0 1 0 1 X 1 0	16 words
Pr_RwWF_Coherent_Excl_32w	1 X 0 0 0 0 1 0 1 X 1 1	32 words
Pr_RwWF	1 X 0 0 0 0 1 X X X X X †	Read with write forthcoming
Pr_RwWF_Coherent	1 X 0 0 0 0 1 0 X X X X †	Coherent block
Pr_RwWF_Coherent_Link	1 X 0 0 0 0 1 0 X 1 X X †	Link address retained
Pr_RwWF_Coherent_Link_4w	1 X 0 0 0 0 1 0 X 1 0 0	4 words
Pr_RwWF_Coherent_Link_8w	1 X 0 0 0 0 1 0 X 1 0 1	8 words
Pr_RwWF_Coherent_Link_16w	1 X 0 0 0 0 1 0 X 1 1 0	16 words
Pr_RwWF_Coherent_Link_32w	1 X 0 0 0 0 1 0 X 1 1 1	32 words
Pr_RwWF_Coherent_NoLink	1 X 0 0 0 0 1 0 X 0 X X †	Link address not retained
Pr_RwWF_Coherent_4w	1 X 0 0 0 0 1 0 X X 0 0	4 words
Pr_RwWF_Coherent_8w	1 X 0 0 0 0 1 0 X X 0 1	8 words
Pr_RwWF_Coherent_16w	1 X 0 0 0 0 1 0 X X 1 0	16 words
Pr_RwWF_Coherent_32w	1 X 0 0 0 0 1 0 X X 1 1	32 words
Pr_RwWF	1 X 0 0 0 0 1 X X X X X †	Read with write forthcoming
Pr_RwWF_NonCoherent	1 X 0 0 0 0 1 1 0 X X X †	Noncoherent block
Pr_RwWF_NonCoherent_Link	1 X 0 0 0 0 1 1 0 1 X X †	Link address retained
Pr_RwWF_NonCoherent_Link_4w	1 X 0 0 0 0 1 1 0 1 0 0	4 words
Pr_RwWF_NonCoherent_Link_8w	1 X 0 0 0 0 1 1 0 1 0 1	8 words
Pr_RwWF_NonCoherent_Link_16w	1 X 0 0 0 0 1 1 0 1 1 0	16 words
Pr_RwWF_NonCoherent_Link_32w	1 X 0 0 0 0 1 1 0 1 1 1	32 words

(Cont.)

Table 2-1 (Cont.)
Control Group Symbol Table (R4000_Ctrl)

Symbol	Control Group Value Reset* ValidIn* SysCmd7 SysCmd2 ValidOut* SysCmd6 SysCmd4 SysCmd1 SysCmd8 SysCmd5 SysCmd3 SysCmd0	Meaning
Pr_RwWF_NonCoherent_NoLink	1 X 0 0 0 0 1 1 0 0 X X †	Link address not retained
Pr_RwWF_NonCoherent _4w	1 X 0 0 0 0 1 1 0 X 0 0	4 words
Pr_RwWF_NonCoherent _8w	1 X 0 0 0 0 1 1 0 X 0 1	8 words
Pr_RwWF_NonCoherent _16w	1 X 0 0 0 0 1 1 0 X 1 0	16 words
Pr_RwWF_NonCoherent _32w	1 X 0 0 0 0 1 1 0 X 1 1	32 words
Pr_RwWF	1 X 0 0 0 0 1 X X X X X †	Read with write forthcoming
RESERVED=>Pr_RwWF-3	1 X 0 0 0 0 1 1 1 X X X	Reserved; illegal
Pr_Read	1 X 0 0 0 0 X X X X X X †	Read
Pr_Read_Coherent_Excl	1 X 0 0 0 0 X 0 1 X X X †	Coherent block, exclusive
Pr_Read_Coherent_Excl_Link	1 X 0 0 0 0 X 0 1 1 X X †	Link address retained
Pr_Read_Coherent_Excl_Link_4w	1 X 0 0 0 0 X 0 1 1 0 0	4 words
Pr_Read_Coherent_Excl_Link_8w	1 X 0 0 0 0 X 0 1 1 0 1	8 words
Pr_Read_Coherent_Excl_Link_16w	1 X 0 0 0 0 X 0 1 1 1 0	16 words
Pr_Read_Coherent_Excl_Link_32w	1 X 0 0 0 0 X 0 1 1 1 1	32 words
Pr_Read_Coherent_Excl_NoLink	1 X 0 0 0 0 X 0 1 0 X X †	Link address not retained
Pr_Read_Coherent_Excl _4w	1 X 0 0 0 0 X 0 1 X 0 0	4 words
Pr_Read_Coherent_Excl _8w	1 X 0 0 0 0 X 0 1 X 0 1	8 words
Pr_Read_Coherent_Excl _16w	1 X 0 0 0 0 X 0 1 X 1 0	16 words
Pr_Read_Coherent_Excl _32w	1 X 0 0 0 0 X 0 1 X 1 1	32 words
Pr_Read	1 X 0 0 0 0 X X X X X X †	Read
Pr_Read_Coherent	1 X 0 0 0 0 X 0 X X X X †	Coherent block
Pr_Read_Coherent_Link	1 X 0 0 0 0 X 0 X 1 X X †	Link address retained
Pr_Read_Coherent_Link_4w	1 X 0 0 0 0 X 0 X 1 0 0	4 words
Pr_Read_Coherent_Link_8w	1 X 0 0 0 0 X 0 X 1 0 1	8 words
Pr_Read_Coherent_Link_16w	1 X 0 0 0 0 X 0 X 1 1 0	16 words
Pr_Read_Coherent_Link_32w	1 X 0 0 0 0 X 0 X 1 1 1	32 words
Pr_Read_Coherent_NoLink	1 X 0 0 0 0 X 0 X 0 X X †	Link address not retained
Pr_Read_Coherent_ _4w	1 X 0 0 0 0 X 0 X X 0 0	4 words
Pr_Read_Coherent_ _8w	1 X 0 0 0 0 X 0 X X 0 1	8 words
Pr_Read_Coherent_ _16w	1 X 0 0 0 0 X 0 X X 1 0	16 words
Pr_Read_Coherent_ _32w	1 X 0 0 0 0 X 0 X X 1 1	32 words
Pr_Read	1 X 0 0 0 0 X X X X X X †	Read
Pr_Read_NonCoherent	1 X 0 0 0 0 X 1 0 X X X †	Noncoherent block
Pr_Read_NonCoherent_Link	1 X 0 0 0 0 X 1 0 1 X X †	Link address retained
Pr_Read_NonCoherent_Link_4w	1 X 0 0 0 0 X 1 0 1 0 0	4 words
Pr_Read_NonCoherent_Link_8w	1 X 0 0 0 0 X 1 0 1 0 1	8 words
Pr_Read_NonCoherent_Link_16w	1 X 0 0 0 0 X 1 0 1 1 0	16 words
Pr_Read_NonCoherent_Link_32w	1 X 0 0 0 0 X 1 0 1 1 1	32 words

(Cont.)

Table 2-1 (Cont.)
Control Group Symbol Table (R4000_Ctrl)

Symbol	Control Group Value Reset* ValidIn* ValidOut* SysCmd8 SysCmd7 SysCmd6 SysCmd5 SysCmd4 SysCmd3 SysCmd2 SysCmd1 SysCmd0	Meaning
Pr_Read_NonCoherent_NoLink	1 X 0 0 0 0 X 1 0 0 X X †	Link address not retained
Pr_Read_NonCoherent _4w	1 X 0 0 0 0 X 1 0 X 0 0	4 words
Pr_Read_NonCoherent _8w	1 X 0 0 0 0 X 1 0 X 0 1	8 words
Pr_Read_NonCoherent _16w	1 X 0 0 0 0 X 1 0 X 1 0	16 words
Pr_Read_NonCoherent _32w	1 X 0 0 0 0 X 1 0 X 1 1	32 words
Pr_Read	1 X 0 0 0 0 X X X X X X †	Read
Pr_Read_Word	1 X 0 0 0 0 0 1 1 X X X †	Double, single, partial word
Pr_Read_1b	1 X 0 0 0 0 0 1 1 0 0 0	One byte valid
Pr_Read_2b	1 X 0 0 0 0 0 1 1 0 0 1	Two bytes valid
Pr_Read_3b	1 X 0 0 0 0 0 1 1 0 1 0	Three bytes valid
Pr_Read_4b	1 X 0 0 0 0 0 1 1 0 1 1	Four bytes valid
Pr_Read_5b	1 X 0 0 0 0 0 1 1 1 0 0	Five bytes valid
Pr_Read_6b	1 X 0 0 0 0 0 1 1 1 0 1	Six bytes valid
Pr_Read_7b	1 X 0 0 0 0 0 1 1 1 1 0	Seven bytes valid
Pr_Read_8b	1 X 0 0 0 0 0 1 1 1 1 1	Eight bytes valid
Pr_Write	1 X 0 0 0 1 0 X X X X X †	Write
RESERVED=>Pr_Write–0	1 X 0 0 0 1 0 0 0 X X X	Reserved; illegal
Pr_Write	1 X 0 0 0 1 0 X X X X X †	Write
RESERVED=>Pr_Write–1	1 X 0 0 0 1 0 0 1 X X X	Reserved; illegal
Pr_Write	1 X 0 0 0 1 0 X X X X X †	Write
Pr_Write_Block	1 X 0 0 0 1 0 1 0 X X X †	Block write
Pr_Write_Replaced	1 X 0 0 0 1 0 1 0 0 X X †	Cache line replaced
Pr_Write_Replaced_4w	1 X 0 0 0 1 0 1 0 0 0 0	4 words
Pr_Write_Replaced_8w	1 X 0 0 0 1 0 1 0 0 0 1	8 words
Pr_Write_Replaced_16w	1 X 0 0 0 1 0 1 0 0 1 0	16 words
Pr_Write_Replaced_32w	1 X 0 0 0 1 0 1 0 0 1 1	32 words
Pr_Write_Retained	1 X 0 0 0 1 0 1 0 1 X X †	Cache line retained
Pr_Write_Retained_4w	1 X 0 0 0 1 0 1 0 1 0 0	4 words
Pr_Write_Retained_8w	1 X 0 0 0 1 0 1 0 1 0 1	8 words
Pr_Write_Retained_16w	1 X 0 0 0 1 0 1 0 1 1 0	16 words
Pr_Write_Retained_32w	1 X 0 0 0 1 0 1 0 1 1 1	32 words
Pr_Write	1 X 0 0 0 1 0 X X X X X †	Write
Pr_Write_Block	1 X 0 0 0 1 0 1 0 X X X †	Block write
Pr_Write_Block_4w	1 X 0 0 0 1 0 1 0 X 0 0 †	4 words
Pr_Write_Block_8w	1 X 0 0 0 1 0 1 0 X 0 1 †	8 words
Pr_Write_Block_16w	1 X 0 0 0 1 0 1 0 X 1 0 †	16 words
Pr_Write_Block_32w	1 X 0 0 0 1 0 1 0 X 1 1 †	32 words

(Cont.)

Table 2-1 (Cont.)
Control Group Symbol Table (R4000_Ctrl)

Symbol	Control Group Value Reset* ValidIn* ValidOut* SysCmd8 SysCmd7 SysCmd6 SysCmd5 SysCmd4 SysCmd3 SysCmd2 SysCmd1 SysCmd0	Meaning
Pr_Write	1 X 0 0 0 1 0 X X X X X †	Write
Pr_Write_Word	1 X 0 0 0 1 0 1 1 X X X †	Double, single, partial word
Pr_Write_1b	1 X 0 0 0 1 0 1 1 0 0 0	One byte valid
Pr_Write_2b	1 X 0 0 0 1 0 1 1 0 0 1	Two bytes valid
Pr_Write_3b	1 X 0 0 0 1 0 1 1 0 1 0	Three bytes valid
Pr_Write_4b	1 X 0 0 0 1 0 1 1 0 1 1	Four bytes valid
Pr_Write_5b	1 X 0 0 0 1 0 1 1 1 0 0	Five bytes valid
Pr_Write_6b	1 X 0 0 0 1 0 1 1 1 0 1	Six bytes valid
Pr_Write_7b	1 X 0 0 0 1 0 1 1 1 1 0	Seven bytes valid
Pr_Write_8b	1 X 0 0 0 1 0 1 1 1 1 1	Eight bytes valid
Pr_Null	1 X 0 0 0 1 1 X X X X X †	Null
Pr_Null_Write	1 X 0 0 0 1 1 0 0 X X X	Null write
RESERVED=>Pr_Null-1	1 X 0 0 0 1 1 0 1 X X X	Reserved; illegal
RESERVED=>Pr_Null-2	1 X 0 0 0 1 1 1 0 X X X	Reserved; illegal
RESERVED=>Pr_Null-3	1 X 0 0 0 1 1 1 1 X X X	Reserved; illegal
Pr_Ivd	1 X 0 0 1 0 0 X X X X X	Invalidate
	1 X 0 0 1 0 0 0 X X X X	Reserved; don't care
	1 X 0 0 1 0 0 1 X X X X	Reserved; don't care
	1 X 0 0 1 0 0 X 0 X X X	Reserved; don't care
	1 X 0 0 1 0 0 X 1 X X X	Reserved; don't care
Pr_Upd	1 X 0 0 1 0 1 X X X X X †	Update
	1 X 0 0 1 0 1 0 X X X X	Reserved; don't care
	1 X 0 0 1 0 1 1 X X X X	Reserved; don't care
Pr_Upd_Compulsory	1 X 0 0 1 0 1 X 0 X X X †	Compulsory
Pr_Upd_Compulsory_1b	1 X 0 0 1 0 1 X 0 0 0 0	One byte valid
Pr_Upd_Compulsory_2b	1 X 0 0 1 0 1 X 0 0 0 1	Two bytes valid
Pr_Upd_Compulsory_3b	1 X 0 0 1 0 1 X 0 0 1 0	Three bytes valid
Pr_Upd_Compulsory_4b	1 X 0 0 1 0 1 X 0 0 1 1	Four bytes valid
Pr_Upd_Compulsory_5b	1 X 0 0 1 0 1 X 0 1 0 0	Five bytes valid
Pr_Upd_Compulsory_6b	1 X 0 0 1 0 1 X 0 1 0 1	Six bytes valid
Pr_Upd_Compulsory_7b	1 X 0 0 1 0 1 X 0 1 1 0	Seven bytes valid
Pr_Upd_Compulsory_8b	1 X 0 0 1 0 1 X 0 1 1 1	Eight bytes valid
Pr_Upd_Potential	1 X 0 0 1 0 1 X 1 X X X †	Potential
Pr_Upd_Potential_1b	1 X 0 0 1 0 1 X 1 0 0 0	One byte valid
Pr_Upd_Potential_2b	1 X 0 0 1 0 1 X 1 0 0 1	Two bytes valid
Pr_Upd_Potential_3b	1 X 0 0 1 0 1 X 1 0 1 0	Three bytes valid
Pr_Upd_Potential_4b	1 X 0 0 1 0 1 X 1 0 1 1	Four bytes valid
Pr_Upd_Potential_5b	1 X 0 0 1 0 1 X 1 1 0 0	Five bytes valid
Pr_Upd_Potential_6b	1 X 0 0 1 0 1 X 1 1 0 1	Six bytes valid
Pr_Upd_Potential_7b	1 X 0 0 1 0 1 X 1 1 1 0	Seven bytes valid

(Cont.)

Table 2-1 (Cont.)
Control Group Symbol Table (R4000_Ctrl)

Symbol	Control Group Value Reset* ValidIn* SysCmd7 SysCmd2 ValidOut* SysCmd6 SysCmd4 SysCmd1 SysCmd8 SysCmd5 SysCmd3 SysCmd0	Meaning
Pr_Upd_Potential_8b	1 X 0 0 1 0 1 X 1 1 1 1	Eight bytes valid
Pr_Upd	1 X 0 0 1 0 1 X X X X X †	Update
Pr_Upd_1b	1 X 0 0 1 0 1 X X 0 0 0 †	One byte valid
Pr_Upd_2b	1 X 0 0 1 0 1 X X 0 0 1 †	Two bytes valid
Pr_Upd_3b	1 X 0 0 1 0 1 X X 0 1 0 †	Three bytes valid
Pr_Upd_4b	1 X 0 0 1 0 1 X X 0 1 1 †	Four bytes valid
Pr_Upd_5b	1 X 0 0 1 0 1 X X 1 0 0 †	Five bytes valid
Pr_Upd_6b	1 X 0 0 1 0 1 X X 1 0 1 †	Six bytes valid
Pr_Upd_7b	1 X 0 0 1 0 1 X X 1 1 0 †	Seven bytes valid
Pr_Upd_8b	1 X 0 0 1 0 1 X X 1 1 1 †	Eight bytes valid
RESERVED=>Pr_Ivtn–X	1 X 0 0 1 1 0 X X X X X	Intervention–Reserved; illegal
RESERVED=>Pr_Snoop–X	1 X 0 0 1 1 1 X X X X X	Snoop–Reserved; illegal
External Commands		
Ex_Read	1 0 X 0 0 0 0 X X X X X †	Read
RESERVED=>Ex_Read–0	1 0 X 0 0 0 0 0 0 X X X	Reserved; illegal
Ex_Read	1 0 X 0 0 0 0 X X X X X †	Read
RESERVED=>Ex_Read–1	1 0 X 0 0 0 0 0 1 X X X	Reserved; illegal
Ex_Read	1 0 X 0 0 0 0 X X X X X †	Read
RESERVED=>Ex_Read–2	1 0 X 0 0 0 0 1 0 X X X	Reserved; illegal
Ex_Read	1 0 X 0 0 0 0 X X X X X †	Read
Ex_Read_Word	1 0 X 0 0 0 0 1 1 X X X †	Double, single, partial word
Ex_Read_1b	1 0 X 0 0 0 0 1 1 0 0 0	One byte valid
Ex_Read_2b	1 0 X 0 0 0 0 1 1 0 0 1	Two bytes valid
Ex_Read_3b	1 0 X 0 0 0 0 1 1 0 1 0	Three bytes valid
Ex_Read_4b	1 0 X 0 0 0 0 1 1 0 1 1	Four bytes valid
Ex_Read_5b	1 0 X 0 0 0 0 1 1 1 0 0	Five bytes valid
Ex_Read_6b	1 0 X 0 0 0 0 1 1 1 0 1	Six bytes valid
Ex_Read_7b	1 0 X 0 0 0 0 1 1 1 1 0	Seven bytes valid
Ex_Read_8b	1 0 X 0 0 0 0 1 1 1 1 1	Eight bytes valid
RESERVED=>Ex_RwWF–X	1 0 X 0 0 0 1 X X X X X	Read with write forthcoming–Reserved; illegal
Ex_Write	1 0 X 0 0 1 0 X X X X X †	Write
RESERVED=>Ex_Write–0	1 0 X 0 0 1 0 0 0 X X X	Reserved; illegal
Ex_Write	1 0 X 0 0 1 0 X X X X X †	Write
RESERVED=>Ex_Write–1	1 0 X 0 0 1 0 0 1 X X X	Reserved; illegal
Ex_Write	1 0 X 0 0 1 0 X X X X X †	Write
RESERVED=>Ex_Write–2	1 0 X 0 0 1 0 1 0 X X X	Reserved; illegal

(Cont.)

Table 2-1 (Cont.)
Control Group Symbol Table (R4000_Ctrl)

Symbol	Control Group Value Reset* ValidIn* ValidOut* SysCmd8 SysCmd7 SysCmd6 SysCmd5 SysCmd4 SysCmd3 SysCmd2 SysCmd1 SysCmd0	Meaning
Ex_Write	1 0 X 0 0 1 0 X X X X X †	Write
Ex_Write_Word	1 0 X 0 0 1 0 1 1 X X X †	Double, single, partial word
Ex_Write_1b	1 0 X 0 0 1 0 1 1 0 0 0	One byte valid
Ex_Write_2b	1 0 X 0 0 1 0 1 1 0 0 1	Two bytes valid
Ex_Write_3b	1 0 X 0 0 1 0 1 1 0 1 0	Three bytes valid
Ex_Write_4b	1 0 X 0 0 1 0 1 1 0 1 1	Four bytes valid
Ex_Write_5b	1 0 X 0 0 1 0 1 1 1 0 0	Five bytes valid
Ex_Write_6b	1 0 X 0 0 1 0 1 1 1 0 1	Six bytes valid
Ex_Write_7b	1 0 X 0 0 1 0 1 1 1 1 0	Seven bytes valid
Ex_Write_8b	1 0 X 0 0 1 0 1 1 1 1 1	Eight bytes valid
Ex_Null	1 0 X 0 0 1 1 X X X X X †	Null
Ex_Null_SI_Release	1 0 X 0 0 1 1 0 0 X X X	System interface release
Ex_Null_SC_Release	1 0 X 0 0 1 1 0 1 X X X	Secondary cache release
RESERVED=>Ex_Null–2	1 0 X 0 0 1 1 1 0 X X X	Reserved; illegal
RESERVED=>Ex_Null–3	1 0 X 0 0 1 1 1 1 X X X	Reserved; illegal
Ex_Ivd	1 0 X 0 1 0 0 X X X X X †	Invalidate
Ex_Ivd_Cancel	1 0 X 0 1 0 0 0 X X X X	Invalidate or update cancel
Ex_Ivd_NoCancel	1 0 X 0 1 0 0 1 X X X X	No cancellation
	1 0 X 0 1 0 0 X 0 X X X	Reserved; don't care
	1 0 X 0 1 0 0 X 1 X X X	Reserved; don't care
Ex_Upd	1 0 X 0 1 0 1 X X X X X †	Update
Ex_Upd _Cancel	1 0 X 0 1 0 1 0 X X X X †	Invalidate or update cancel
Ex_Upd_Shared	1 0 X 0 1 0 1 X 0 X X X †	Cache changed to shared
Ex_Upd_Shared_1b_Cancel	1 0 X 0 1 0 1 0 0 0 0 0	One byte valid
Ex_Upd_Shared_2b_Cancel	1 0 X 0 1 0 1 0 0 0 0 1	Two bytes valid
Ex_Upd_Shared_3b_Cancel	1 0 X 0 1 0 1 0 0 0 1 0	Three bytes valid
Ex_Upd_Shared_4b_Cancel	1 0 X 0 1 0 1 0 0 0 1 1	Four bytes valid
Ex_Upd_Shared_5b_Cancel	1 0 X 0 1 0 1 0 0 1 0 0	Five bytes valid
Ex_Upd_Shared_6b_Cancel	1 0 X 0 1 0 1 0 0 1 0 1	Six bytes valid
Ex_Upd_Shared_7b_Cancel	1 0 X 0 1 0 1 0 0 1 1 0	Seven bytes valid
Ex_Upd_Shared_8b_Cancel	1 0 X 0 1 0 1 0 0 1 1 1	Eight bytes valid
Ex_Upd_Nochange	1 0 X 0 1 0 1 X 1 X X X †	No change to cache state
Ex_Upd_Nochange_1b_Cancel	1 0 X 0 1 0 1 0 1 0 0 0	One byte valid
Ex_Upd_Nochange_2b_Cancel	1 0 X 0 1 0 1 0 1 0 0 1	Two bytes valid
Ex_Upd_Nochange_3b_Cancel	1 0 X 0 1 0 1 0 1 0 1 0	Three bytes valid
Ex_Upd_Nochange_4b_Cancel	1 0 X 0 1 0 1 0 1 0 1 1	Four bytes valid
Ex_Upd_Nochange_5b_Cancel	1 0 X 0 1 0 1 0 1 1 0 0	Five bytes valid
Ex_Upd_Nochange_6b_Cancel	1 0 X 0 1 0 1 0 1 1 0 1	Six bytes valid
Ex_Upd_Nochange_7b_Cancel	1 0 X 0 1 0 1 0 1 1 1 0	Seven bytes valid
Ex_Upd_Nochange_8b_Cancel	1 0 X 0 1 0 1 0 1 1 1 1	Eight bytes valid

(Cont.)

Table 2-1 (Cont.)
Control Group Symbol Table (R4000_Ctrl)

Symbol	Control Group Value Reset* ValidIn* ValidOut* SysCmd8 SysCmd7 SysCmd6 SysCmd5 SysCmd4 SysCmd3 SysCmd2 SysCmd1 SysCmd0	Meaning
Ex_Upd	1 0 X 0 1 0 1 X X X X X †	Update
Ex_Upd _NoCancel	1 0 X 0 1 0 1 1 X X X X †	No cancellation
Ex_Upd_Shared	1 0 X 0 1 0 1 X 0 X X X †	Cache changed to shared
Ex_Upd_Shared_1b	1 0 X 0 1 0 1 X 0 0 0 0	One byte valid
Ex_Upd_Shared_2b	1 0 X 0 1 0 1 X 0 0 0 1	Two bytes valid
Ex_Upd_Shared_3b	1 0 X 0 1 0 1 X 0 0 1 0	Three bytes valid
Ex_Upd_Shared_4b	1 0 X 0 1 0 1 X 0 0 1 1	Four bytes valid
Ex_Upd_Shared_5b	1 0 X 0 1 0 1 X 0 1 0 0	Five bytes valid
Ex_Upd_Shared_6b	1 0 X 0 1 0 1 X 0 1 0 1	Six bytes valid
Ex_Upd_Shared_7b	1 0 X 0 1 0 1 X 0 1 1 0	Seven bytes valid
Ex_Upd_Shared_8b	1 0 X 0 1 0 1 X 0 1 1 1	Eight bytes valid
Ex_Upd_Nochange	1 0 X 0 1 0 1 X 1 X X X †	No change to cache state
Ex_Upd_Nochange_1b	1 0 X 0 1 0 1 X 1 0 0 0	One byte valid
Ex_Upd_Nochange_2b	1 0 X 0 1 0 1 X 1 0 0 1	Two bytes valid
Ex_Upd_Nochange_3b	1 0 X 0 1 0 1 X 1 0 1 0	Three bytes valid
Ex_Upd_Nochange_4b	1 0 X 0 1 0 1 X 1 0 1 1	Four bytes valid
Ex_Upd_Nochange_5b	1 0 X 0 1 0 1 X 1 1 0 0	Five bytes valid
Ex_Upd_Nochange_6b	1 0 X 0 1 0 1 X 1 1 0 1	Six bytes valid
Ex_Upd_Nochange_7b	1 0 X 0 1 0 1 X 1 1 1 0	Seven bytes valid
Ex_Upd_Nochange_8b	1 0 X 0 1 0 1 X 1 1 1 1	Eight bytes valid
Ex_Upd	1 0 X 0 1 0 1 X X X X X †	Update
Ex_Upd_1b	1 0 X 0 1 0 1 X X 0 0 0 †	One byte valid
Ex_Upd_2b	1 0 X 0 1 0 1 X X 0 0 1 †	Two bytes valid
Ex_Upd_3b	1 0 X 0 1 0 1 X X 0 1 0 †	Three bytes valid
Ex_Upd_4b	1 0 X 0 1 0 1 X X 0 1 1†	Four bytes valid
Ex_Upd_5b	1 0 X 0 1 0 1 X X 1 0 0 †	Five bytes valid
Ex_Upd_6b	1 0 X 0 1 0 1 X X 1 0 1 †	Six bytes valid
Ex_Upd_7b	1 0 X 0 1 0 1 X X 1 1 0 †	Seven bytes valid
Ex_Upd_8b	1 0 X 0 1 0 1 X X 1 1 1†	Eight bytes valid
Ex_Ivtn	1 0 X 0 1 1 0 X X X X X †	Intervention
Ex_Ivtn _Cancel	1 0 X 0 1 1 0 0 X X X X †	Invalidate or update cancel
Ex_Ivtn_Rtn_Drty	1 0 X 0 1 1 0 X 0 X X X †	Return cache line if dirty
Ex_Ivtn_Rtn_Drty_Chg0_Cancel	1 0 X 0 1 1 0 0 0 0 0 0	No change to cache state
Ex_Ivtn_Rtn_Drty_Chg1_Cancel	1 0 X 0 1 1 0 0 0 0 0 1	Cache state change 1‡
Ex_Ivtn_Rtn_Drty_Chg2_Cancel	1 0 X 0 1 1 0 0 0 0 1 0	Cache state change 2‡
Ex_Ivtn_Rtn_Drty_Chg3_Cancel	1 0 X 0 1 1 0 0 0 0 1 1	Cache state change 3‡
Ex_Ivtn_Rtn_Drty_Chg4_Cancel	1 0 X 0 1 1 0 0 0 1 0 0	Cache state change 4‡
Ex_Ivtn_Rtn_Drty_Chg5_Cancel	1 0 X 0 1 1 0 0 0 1 0 1	Change cache to invalid
RESERVED=>Ex_Ivtn–6	1 0 X 0 1 1 0 X X 1 1 0	Reserved; illegal
RESERVED=>Ex_Ivtn–7	1 0 X 0 1 1 0 X X 1 1 1	Reserved; illegal

(Cont.)

Table 2-1 (Cont.)
Control Group Symbol Table (R4000_Ctrl)

Symbol	Control Group Value Reset* ValidIn* ValidOut* SysCmd8 SysCmd7 SysCmd6 SysCmd5 SysCmd4 SysCmd3 SysCmd2 SysCmd1 SysCmd0	Meaning
Ex_Ivtn_Rtn_Excl	1 0 X 0 1 1 0 X 1 X X X †	Return cache line if exclusive
Ex_Ivtn_Rtn_Excl_Chg0_Cancel	1 0 X 0 1 1 0 0 1 0 0 0	No change to cache state
Ex_Ivtn_Rtn_Excl_Chg1_Cancel	1 0 X 0 1 1 0 0 1 0 0 1	Cache state change 1‡
Ex_Ivtn_Rtn_Excl_Chg2_Cancel	1 0 X 0 1 1 0 0 1 0 1 0	Cache state change 2‡
Ex_Ivtn_Rtn_Excl_Chg3_Cancel	1 0 X 0 1 1 0 0 1 0 1 1	Cache state change 3‡
Ex_Ivtn_Rtn_Excl_Chg4_Cancel	1 0 X 0 1 1 0 0 1 1 0 0	Cache state change 4‡
Ex_Ivtn_Rtn_Excl_Chg5_Cancel	1 0 X 0 1 1 0 0 1 1 0 1	Change cache to invalid
RESERVED=>Ex_Ivtn–6	1 0 X 0 1 1 0 X X 1 1 0	Reserved; illegal
RESERVED=>Ex_Ivtn–7	1 0 X 0 1 1 0 X X 1 1 1	Reserved; illegal
Ex_Ivtn	1 0 X 0 1 1 0 X X X X X †	Intervention
Ex_Ivtn _NoCancel	1 0 X 0 1 1 0 1 X X X X †	No cancellation
Ex_Ivtn_Rtn_Drty	1 0 X 0 1 1 0 X 0 X X X †	Return cache line if dirty
Ex_Ivtn_Rtn_Drty_Chg0	1 0 X 0 1 1 0 X 0 0 0 0	No change to cache state
Ex_Ivtn_Rtn_Drty_Chg1	1 0 X 0 1 1 0 X 0 0 0 1	Cache state change 1‡
Ex_Ivtn_Rtn_Drty_Chg2	1 0 X 0 1 1 0 X 0 0 1 0	Cache state change 2‡
Ex_Ivtn_Rtn_Drty_Chg3	1 0 X 0 1 1 0 X 0 0 1 1	Cache state change 3‡
Ex_Ivtn_Rtn_Drty_Chg4	1 0 X 0 1 1 0 X 0 1 0 0	Cache state change 4‡
Ex_Ivtn_Rtn_Drty_Chg5	1 0 X 0 1 1 0 X 0 1 0 1	Change cache to invalid
RESERVED=>Ex_Ivtn–6	1 0 X 0 1 1 0 X X 1 1 0	Reserved; illegal
RESERVED=>Ex_Ivtn–7	1 0 X 0 1 1 0 X X 1 1 1	Reserved; illegal
Ex_Ivtn_Rtn_Excl	1 0 X 0 1 1 0 X 1 X X X †	Return cache line if exclusive
Ex_Ivtn_Rtn_Excl_Chg0	1 0 X 0 1 1 0 X 1 0 0 0	No change to cache state
Ex_Ivtn_Rtn_Excl_Chg1	1 0 X 0 1 1 0 X 1 0 0 1	Cache state change 1‡
Ex_Ivtn_Rtn_Excl_Chg2	1 0 X 0 1 1 0 X 1 0 1 0	Cache state change 2‡
Ex_Ivtn_Rtn_Excl_Chg3	1 0 X 0 1 1 0 X 1 0 1 1	Cache state change 3‡
Ex_Ivtn_Rtn_Excl_Chg4	1 0 X 0 1 1 0 X 1 1 0 0	Cache state change 4‡
Ex_Ivtn_Rtn_Excl_Chg5	1 0 X 0 1 1 0 X 1 1 0 1	Change cache to invalid
RESERVED=>Ex_Ivtn–6	1 0 X 0 1 1 0 X X 1 1 0	Reserved; illegal
RESERVED=>Ex_Ivtn–7	1 0 X 0 1 1 0 X X 1 1 1	Reserved; illegal
Ex_Ivtn	1 0 X 0 1 1 0 X X X X X †	Intervention
Ex_Ivtn_Chg0	1 0 X 0 1 1 0 X X 0 0 0 †	No change to cache state
Ex_Ivtn_Chg1	1 0 X 0 1 1 0 X X 0 0 1 †	Cache state change 1‡
Ex_Ivtn_Chg2	1 0 X 0 1 1 0 X X 0 1 0 †	Cache state change 2‡
Ex_Ivtn_Chg3	1 0 X 0 1 1 0 X X 0 1 1 †	Cache state change 3‡
Ex_Ivtn_Chg4	1 0 X 0 1 1 0 X X 1 0 0 †	Cache state change 4‡
Ex_Ivtn_Chg5	1 0 X 0 1 1 0 X X 1 0 1 †	Change cache to invalid
RESERVED=>Ex_Ivtn–6	1 0 X 0 1 1 0 X X 1 1 0	Reserved; illegal
RESERVED=>Ex_Ivtn–7	1 0 X 0 1 1 0 X X 1 1 1	Reserved; illegal

(Cont.)

Table 2-1 (Cont.)
Control Group Symbol Table (R4000_Ctrl)

Symbol	Control Group Value Reset* ValidIn* ValidOut* SysCmd8 SysCmd7 SysCmd6 SysCmd5 SysCmd4 SysCmd3 SysCmd2 SysCmd1 SysCmd0	Meaning
Ex_Snoop	1 0 X 0 1 1 1 X X X X X †	Snoop
Ex_Snoop _Cancel	1 0 X 0 1 1 1 0 X X X X †	Invalidate or update cancel
Ex_Snoop_Chg0_Cancel	1 0 X 0 1 1 1 0 X 0 0 0	No change to cache state
Ex_Snoop_Chg1_Cancel	1 0 X 0 1 1 1 0 X 0 0 1	Cache state change 1‡
Ex_Snoop_Chg2_Cancel	1 0 X 0 1 1 1 0 X 0 1 0	Cache state change 2‡
Ex_Snoop_Chg3_Cancel	1 0 X 0 1 1 1 0 X 0 1 1	Cache state change 3‡
Ex_Snoop_Chg4_Cancel	1 0 X 0 1 1 1 0 X 1 0 0	Cache state change 4‡
Ex_Snoop_Chg5_Cancel	1 0 X 0 1 1 1 0 X 1 0 1	Change cache to invalid
RESERVED=>Ex_Snoop–6	1 0 X 0 1 1 1 X X 1 1 0	Reserved; illegal
RESERVED=>Ex_Snoop–7	1 0 X 0 1 1 1 X X 1 1 1	Reserved; illegal
	1 0 X 0 1 1 1 X 0 X X X	Reserved; don't care
	1 0 X 0 1 1 1 X 1 X X X	Reserved; don't care
Ex_Snoop	1 0 X 0 1 1 1 X X X X X †	Snoop
Ex_Snoop _NoCancel	1 0 X 0 1 1 1 1 X X X X †	No cancellation
Ex_Snoop_Chg0	1 0 X 0 1 1 1 X X 0 0 0 †	No change to cache state
Ex_Snoop_Chg1	1 0 X 0 1 1 1 X X 0 0 1 †	Cache state change 1‡
Ex_Snoop_Chg2	1 0 X 0 1 1 1 X X 0 1 0 †	Cache state change 2‡
Ex_Snoop_Chg3	1 0 X 0 1 1 1 X X 0 1 1 †	Cache state change 3‡
Ex_Snoop_Chg4	1 0 X 0 1 1 1 X X 1 0 0 †	Cache state change 4‡
Ex_Snoop_Chg5	1 0 X 0 1 1 1 X X 1 0 1 †	Change cache to invalid
RESERVED=>Ex_Snoop–6	1 0 X 0 1 1 1 X X 1 1 0	Reserved; illegal
RESERVED=>Ex_Snoop–7	1 0 X 0 1 1 1 X X 1 1 1	Reserved; illegal
	1 0 X 0 1 1 1 X 0 X X X	Reserved; don't care
	1 0 X 0 1 1 1 X 1 X X X	Reserved; don't care
Special Purpose Symbols		
Ex_Cmd_Cancel	1 0 X 0 1 X X 0 X X X X †	Invalid or update cancel
Ex_Cmd_NoCancel	1 0 X 0 1 X X 1 X X X X †	No cancellation
Any Commands		
Read	1 X X 0 0 0 X X X X X X †	Read
RwWF	1 X X 0 0 0 1 X X X X X †	Read with write forthcoming
Write	1 X X 0 0 1 0 X X X X X †	Write
Null	1 X X 0 0 1 1 X X X X X †	Null
Ivd	1 X X 0 1 0 0 X X X X X †	Invalidate
Upd	1 X X 0 1 0 1 X X X X X †	Update
Ivtn	1 X X 0 1 1 0 X X X X X †	Intervention
Snoop	1 X X 0 1 1 1 X X X X X †	Snoop

(Cont.)

Table 2-1 (Cont.)
Control Group Symbol Table (R4000_Ctrl)

Symbol	Control Group Value Reset* ValidIn* ValidOut* SysCmd8 SysCmd7 SysCmd6 SysCmd5 SysCmd4 SysCmd3 SysCmd2 SysCmd1 SysCmd0	Meaning
Processor Data Identifiers		
Pr_Data_Error_EOD	1 X 0 1 0 X 1 X X X X X	Erroneous data & end of data
Pr_Data_Error	1 X 0 1 X X 1 X X X X X	Erroneous data
Pr_Data_Resp_EOD	1 X 0 1 0 0 X X X X X X	Response data & end of data
Pr_Data_Resp	1 X 0 1 X 0 X X X X X X	Response data
Pr_Data_EOD	1 X 0 1 0 X X X X X X X	Data & end of data
Pr_Data	1 X 0 1 X X X X X X X X	Data
	1 X 0 1 0 X X X X X X X	Last data element
	1 X 0 1 1 X X X X X X X	Not last data element
	1 X 0 1 X 0 X X X X X X	Response data
	1 X 0 1 X 1 X X X X X X	Not response data
	1 X 0 1 X X 0 X X X X X	Error free
	1 X 0 1 X X 1 X X X X X	Erroneous
	1 X 0 1 X X X 0 X X X X	Reserved; don't care
	1 X 0 1 X X X 1 X X X X	Reserved; don't care
	1 X 0 1 X X X X 0 X X X	Reserved; don't care
	1 X 0 1 X X X X 1 X X X	Reserved; don't care
	1 X 0 1 X X X X X X X X	Noncoherent data identifiers– Reserved; don't care
	1 X 0 1 X X X X X 0 0 0	Coherent data identifiers– Cache invalid
	1 X 0 1 X X X X X 0 0 1	Reserved; illegal
	1 X 0 1 X X X X X 0 1 0	Reserved; illegal
	1 X 0 1 X X X X X 0 1 1	Reserved; illegal
	1 X 0 1 X X X X X 1 0 0	Cache clean exclusive
	1 X 0 1 X X X X X 1 0 1	Cache dirty exclusive
	1 X 0 1 X X X X X 1 1 0	Cache shared
	1 X 0 1 X X X X X 1 1 1	Cache dirty shared
External Data Identifiers		
Ex_Data_Error_EOD	1 0 X 1 0 X 1 X X X X X	Erroneous data & end of data
Ex_Data_Error	1 0 X 1 X X 1 X X X X X	Erroneous data
Ex_Data_Resp_EOD	1 0 X 1 0 0 X X X X X X	Response data & end of data
Ex_Data_Resp	1 0 X 1 X 0 X X X X X X	Response data
Ex_Data_EOD	1 0 X 1 0 X X X X X X X	Data & end of data
Ex_Data	1 0 X 1 X X X X X X X X	Data
	1 0 X 1 0 X X X X X X X	Last data element
	1 0 X 1 1 X X X X X X X	Not last data element

(Cont.)

Table 2-1 (Cont.)
Control Group Symbol Table (R4000_Ctrl)

Symbol	Control Group Value Reset* ValidIn* ValidOut* SysCmd8 SysCmd7 SysCmd6 SysCmd5 SysCmd4 SysCmd3 SysCmd2 SysCmd1 SysCmd0	Meaning
	1 0 X 1 X 0 X X X X X X	Response data
	1 0 X 1 X 1 X X X X X X	Not response data
	1 0 X 1 X X 0 X X X X X	Error free
	1 0 X 1 X X 1 X X X X X	Erroneous
	1 0 X 1 X X X 0 X X X X	Check the data
	1 0 X 1 X X X 1 X X X X	Don't check the data
	1 0 X 1 X X X X 0 X X X	Reserved; don't care
	1 0 X 1 X X X X 1 X X X	Reserved; don't care
	1 0 X 1 X X X X X X X X	Noncoherent data identifiers– Reserved; don't care
	1 0 X 1 X X X X X 0 0 0	Coherent data identifiers– Cache invalid
	1 0 X 1 X X X X X 0 0 1	Reserved; illegal
	1 0 X 1 X X X X X 0 1 0	Reserved; illegal
	1 0 X 1 X X X X X 0 1 1	Reserved; illegal
	1 0 X 1 X X X X X 1 0 0	Cache clean exclusive
	1 0 X 1 X X X X X 1 0 1	Cache dirty exclusive
	1 0 X 1 X X X X X 1 1 0	Cache shared
	1 0 X 1 X X X X X 1 1 1	Cache dirty shared
Any Data Identifiers		
Data_Error_EOD	1 X X 1 0 X 1 X X X X X †	Erroneous data & end of data
Data_Error	1 X X 1 X X 1 X X X X X †	Erroneous data
Data_Resp_EOD	1 X X 1 0 0 X X X X X X †	Response data & end of data
Data_Resp	1 X X 1 X 0 X X X X X X †	Response data
Data_EOD	1 X X 1 0 X X X X X X X †	Data & end of data
Data	1 X X 1 X X X X X X X X †	Data
Any Cycles		
Pr_Cmd	1 X 0 0 X X X X X X X X †	Processor command cycle
Ex_Cmd	1 0 X 0 X X X X X X X X †	External command cycle
Cmd	1 X X 0 X X X X X X X X †	Any command cycle
Pr_Data	1 X 0 1 X X X X X X X X †	Processor data cycle
Ex_Data	1 0 X 1 X X X X X X X X †	External data cycle
Data	1 X X 1 X X X X X X X X †	Any data cycle
Pr	1 X 0 X X X X X X X X X †	Processor cycle (cmd or data)
Ex	1 0 X X X X X X X X X X †	External cycle (cmd or data)
	1 X X X X X X X X X X X	Any cycle (cmd or data)

† These symbols are only for triggering purposes; they are not displayed.
‡ Refer to Table 2-2 for defintions of cache state change.

Table 2-2 shows the definitions of the cache state change symbols from the previous table.

Table 2-2
Cache State Change Definitions

Symbol Definition	Meaning
Cache state change 1	If cache state is clean exclusive, change to shared, otherwise no change to cache state
Cache state change 2	If cache state is exclusive or shared, change to invalid, otherwise no change to cache state
Cache state change 3	If cache state is clean exclusive, change to shared or if cache state is dirty exclusive, change to dirty shared, otherwise no change to cache state
Cache state change 4	If cache state is clean exclusive, dirty exclusive, or dirty shared, change to shared, otherwise no change to cache state

Table 2-3 shows the name, bit pattern, and meaning for the symbols in the file R4000_Intr, the Interrupt group symbol table.

Table 2-3
Interrupt Group Symbol Table (R4000_Intr)

Symbol	Intr Group Value Int*_D NMI*	Meaning
–	1 1	No interrupt
NMI†	X 0	Non-maskable interrupt
INT†	0 X	Any other interrupt
† If both interrupts occur, only the NMI will be displayed.		

Table 2-4 shows the name, bit pattern, and meaning for the symbols in the file R4000_Ack, the Acknowledge group symbol table.

Table 2-4
Acknowledge Group Symbol Table (R4000_Ack)

Symbol	Ack Group Value IvdErr* IvdAck*	Meaning
–	1 1	No invalidate or update acknowledge
ERR	0 X	Unsuccessful completion of a processor invalidate or update request
ACK	X 0	Successful completion of a processor invalidate or update request

Copying and Editing the Predefined Symbol Tables

You cannot directly edit any symbol tables supplied by microprocessor support. But you can make a copy of a predefined symbol table and then edit the copy for your specific use.

To create a new symbol table, follow these steps:

1. Select the Symbol Editor menu.
2. Press F2: FILE FUNCTIONS.
3. Select Open File in the Function field and press Return.
4. Select New File in the Edit Status field and press Return.
5. Enter a new symbol table file name in the New File Name field.
6. Select Pattern in the Table Type field to match the symbol table you are copying and press Return.
7. Press F5: EXECUTE FUNCTION.
8. Select Merge Files in the Function field and press Return.
9. Select the file to base your new symbol table on, such as the R4000_Ctrl file and press Return.
10. Press F5: EXECUTE FUNCTION.
11. Press F8: EXIT & SAVE.
12. Edit the file as desired keeping the following in mind:
    - If the new symbol has fewer don't cares than an existing symbol, it must be placed ahead of the existing symbol.
    - If the new symbol has more don't cares than an existing symbol, it must be placed after the existing symbol.
    - If the new symbol is only used for triggering and not for display, it must be placed after all RESERVED symbols. Symbols placed ahead of RESERVED symbols can be used for either triggering or display.
    - Do not duplicate symbol names.

    Also refer to your *DAS 9200 System User Manual* for more information on editing the symbol table.
13. Select the Channel menu.
14. Change the file name of the symbol table for the Control group (or whichever group's symbol table you are replacing) to the one that you specified in step 5.

LABELS

The probe connectors, cables, and interface housings for the 92A96 Module have color-coded labels. Table 2-5 shows the color of both the probe connector and interface housing labels, as well as the module section and clock assignments.

Table 2-5
Label Information

Label Color	Sections	Clock
Orange	A0, A1, C0	Ck 0
Green	A2, A3, C1	Ck 1
Blue	D0, D1, C2	Ck 2
Gray	D2, D3, C3	Ck 3

Each interface housing connects to three 8-channel probes and a single clock probe. Individual 8-channel probes are labeled with ground and color-coded channel assignments (7-0) only as shown in Figure 2-1.

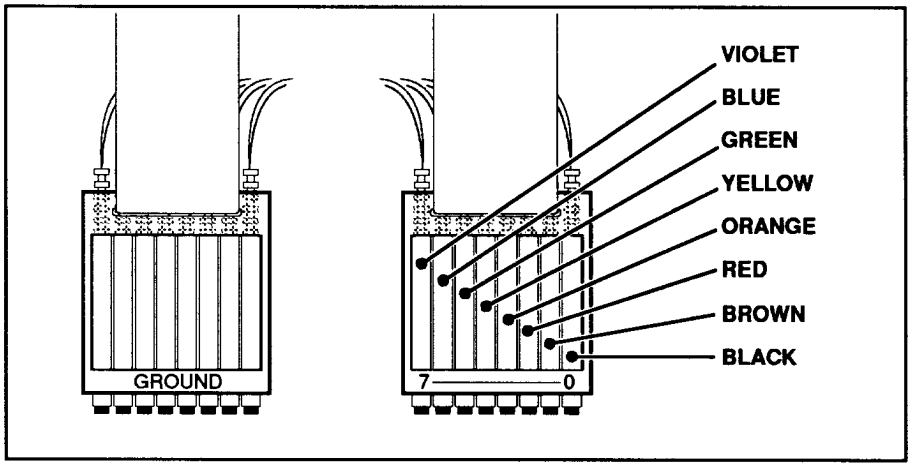


Figure 2-1. Probe channel color and labels on an 8-channel probe.

You should apply slot number labels as shown in Figure 2-2 if there is more than one 92A96 Module in the DAS 9200. These slot numbers will help you identify which module is connected to the probe adapter in a multimodule system.

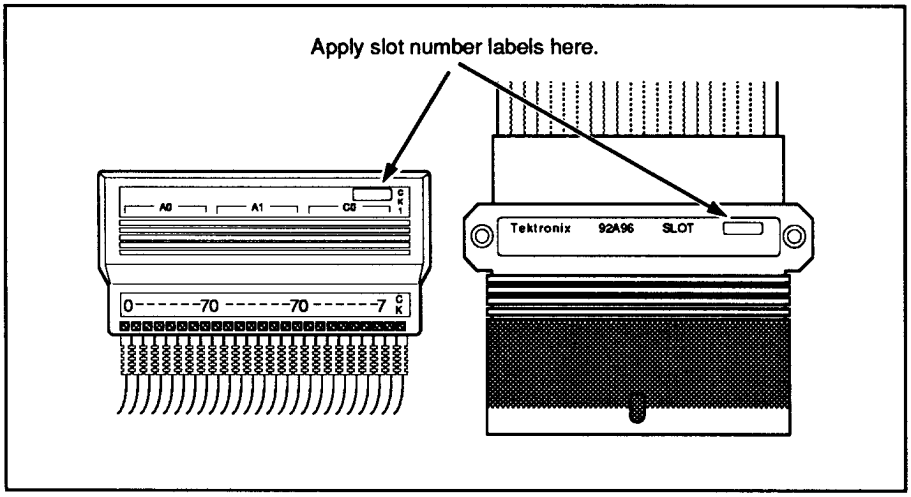


Figure 2-2. Applying slot number labels.

CONFIGURING THE PROBE ADAPTER

There are three jumpers on the probe adapter. One is used to configure the probe adapter for either normal operation or to acquire timing information. The other two are used to calibrate the timing of the TClock signal on the probe adapter when the TClock signal of the R4000 system is buffered.

Normal/Timing Jumper

The Normal/Timing jumper, J1411, should be placed in the Normal position to acquire state or disassembly data, and in the Timing position to acquire timing data. Table 2-6 shows how to position this jumper depending on the type of clocking you're using and the type of display you want to view.

Table 2-6
Normal/Timing Jumper Information

J1411 Position	Clocking	Display Menu
Normal	Custom	Disassembly, State ,or Graph
	External	State or Graph
Timing	Internal	Timing
	External	Timing or State

Figure 2-3 shows the location of J1411 on the probe adapter.

TClock0/TClock1 Jumpers

If your system buffers the TClock signal, the TClock signal on the probe adapter needs to be realigned with the data transitions occurring in the R4000 system. The TClock signal on the probe adapter should occur from 0 to 1 ns after the buffered TClock signal in the system. Measure the difference between the CK3 podlet square pin on the probe adapter and the buffered TClock in the R4000 system using a suitable bandwidth oscilloscope. (Do not measure from the TClock signal of the R4000 microprocessor.) Move both J1412 (TClock0 jumper) and J1421 (TClock1 jumper) to the same position to attain this delay.

If your system does not buffer the TClock signal, do not move either of these jumpers from the N position (pins 1 and 2).

Figure 2-3 shows the location of J1412 and J1421 on the probe adapter.

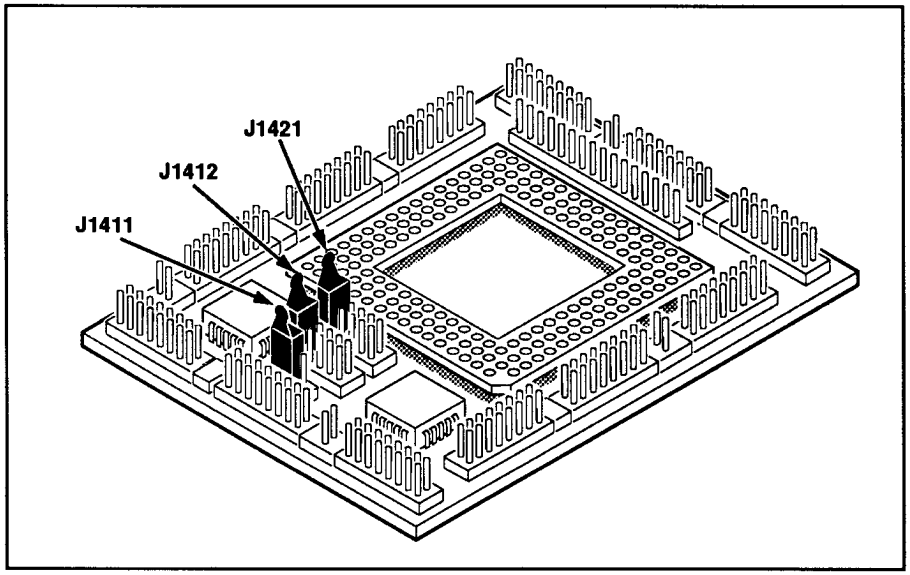


Figure 2-3. Jumper locations on the probe adapter.

CONNECTING THE PROBE ADAPTER

Before acquiring data, you must insert the microprocessor, and connect the probe adapter to both the clock and 8-channel probes for the 92A96 Module being used, and to the SUT. Your R4000 system must have a minimum amount of clear space surrounding the R4000 microprocessor to accommodate the probe adapter. Figures C-1 and C-2 in *Appendix C: Service Information* shows these dimensions.

This discussion describes how to connect the 92A96 Module to the R4000 system.

CAUTION

Static discharge can damage the microprocessor, probe adapter, 8-channel probes, clock probes, or the 92A96 Module. To prevent static damage, observe the following precautions while following all connection procedures.

Handle the microprocessor only in a static-free environment.

To discharge your stored static electricity, grasp the ground lug located on the back of the DAS 9200. Then, touch any of the ground pins (the row of pins closest to the edge of the probe adapter circuit board) to discharge stored static electricity from the probe adapter.

Always wear a grounding wrist strap, or similar device, while handling the microprocessor and probe adapter.

The 92A96 probe cables should already be connected to the probe connectors. If not, refer to your acquisition module user manual for the connection procedures.

Placing the Microprocessor on the Probe Adapter

To place the microprocessor on the probe adapter, follow this procedure:

1. Power down your R4000 system and carefully remove the microprocessor. (It is not necessary to power down the DAS 9200.)
2. Use the antistatic shipping material to support the probe adapter, as shown in Figure 2-5, while installing the microprocessor and connecting the clock and 8-channel probes. This prevents the circuit board from being flexed and the socket pins from being bent.
3. If using a Thermalloy PGA E-Z Mount heat sink with the standard (R4000PC) probe adapter, you must install the extra replacement socket on top of the probe adapter before plugging in the R4000 microprocessor. This allows clearance between the heat sink and J1281.

If you are using the option 2S (R4000SC/R4000MC) probe adapter, you do not need to install the extra replacement socket on top of the probe adapter.

4. Before plugging the R4000 microprocessor into the probe adapter, be sure that the frame or shoes of the Thermalloy PGA E-Z Mount are oriented properly. (The Thermalloy mount attaches the heat sink to the R4000 microprocessor.) This allows clearance for the 92A96 clock and 8-channel probes. Figure 2-4 shows how to orient the frame or shoes.

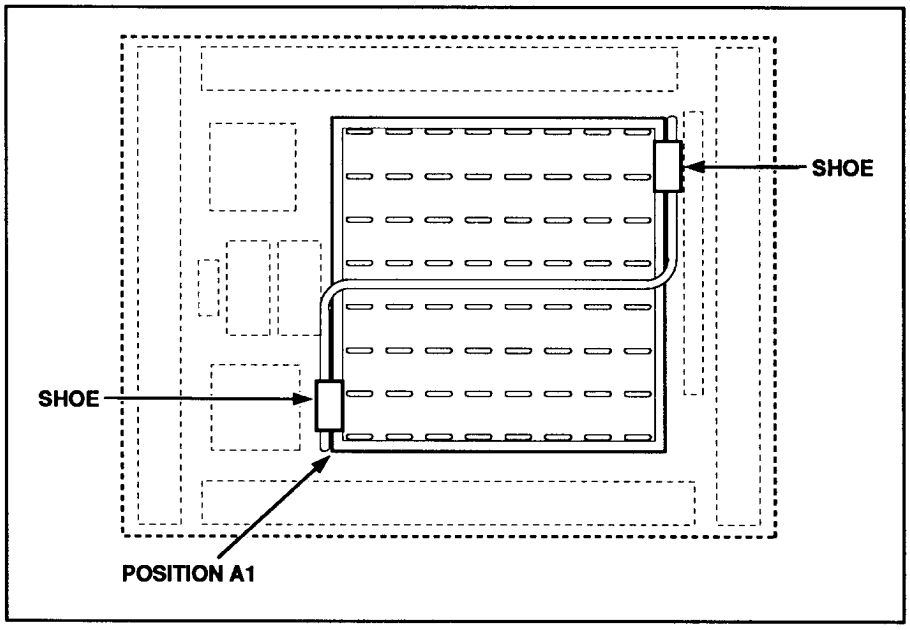


Figure 2-4. Orienting the frame or shoes to the R4000 microprocessor.

5. Carefully plug the R4000 microprocessor with the heat sink attached into the probe adapter socket. Align position A1 on the R4000 to position A1 of the probe adapter socket.

CAUTION

The R4000 microprocessor can be permanently damaged if you insert it with the wrong orientation in the socket on the probe adapter. The probe adapter has an arrow that shows how to properly orient position A1 of the R4000 microprocessor when installing it in the probe adapter socket. Figure 2-5 shows the proper alignment of position A1 on the microprocessor and the probe adapter.

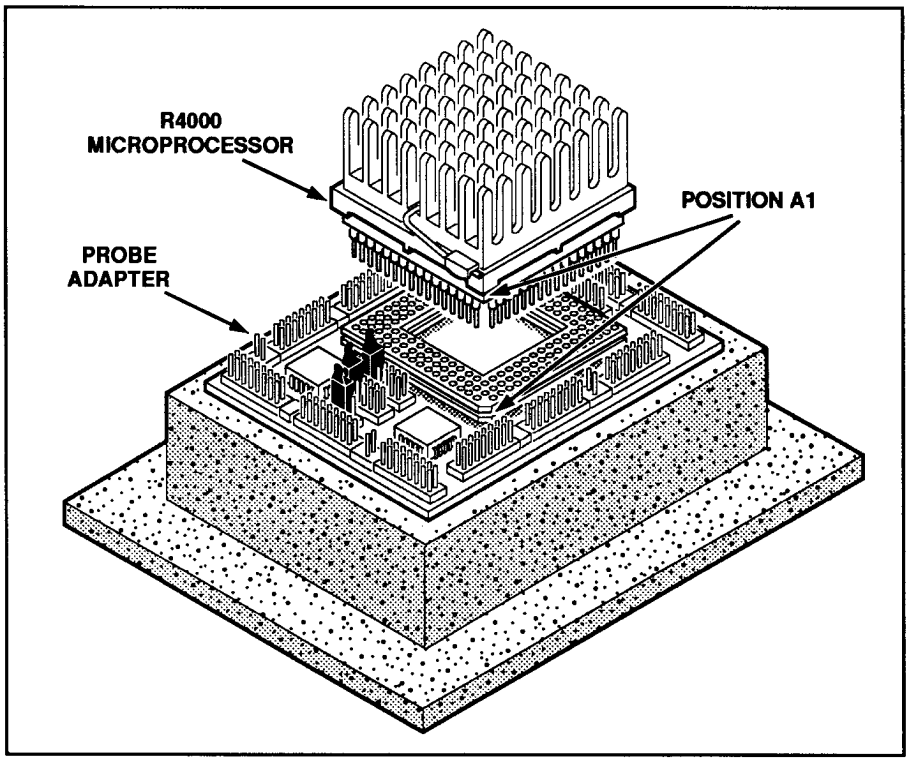


Figure 2-5. Placing the microprocessor on the probe adapter.

Connecting the Probes to the Probe Adapter

To connect the 92A96 Module clocks and 8-channel probes to the probe adapter, refer to Figures 2-6 and 2-7, and use the following procedure. Refer to Appendix C for information on removing and replacing the 8-channel probe podlets.

1. Use the antistatic shipping material to support the probe adapter while connecting the clock and 8-channel probes, as shown in Figure 2-6. This prevents the circuit board from being flexed and the socket pins from being bent.
2. Match the section names and channel numbers on the interface housing to the probe adapter. Figure 2-6 shows how to connect the probes.

CAUTION

Connect only the signal connectors to the signal pins and the ground connectors to the ground pins. Connecting any R4000 signal to a probe adapter ground pin can damage the microprocessor.

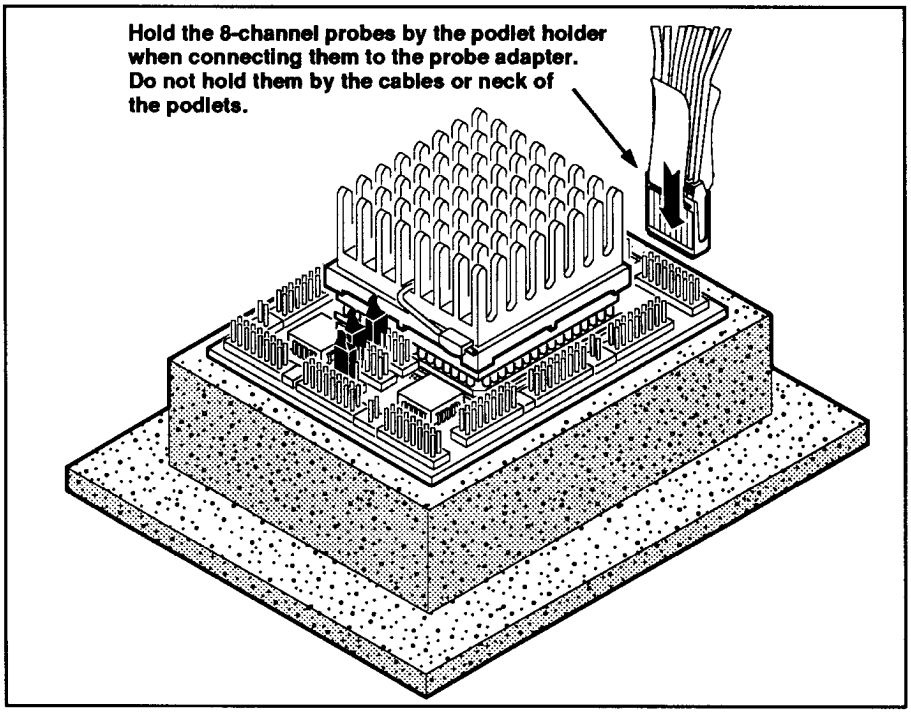


Figure 2-6. Connecting the probes to the probe adapter.

3. Connect the clocks and 8-channel probes to the appropriate sets of square pins on the probe adapter as shown in Figure 2-7. The signal connector is on the color-coded side of the podlet; the ground connector is on the opposite side of the connector and is labeled Ground (or GND). All square pins closest to the edge of the circuit board connect to ground.

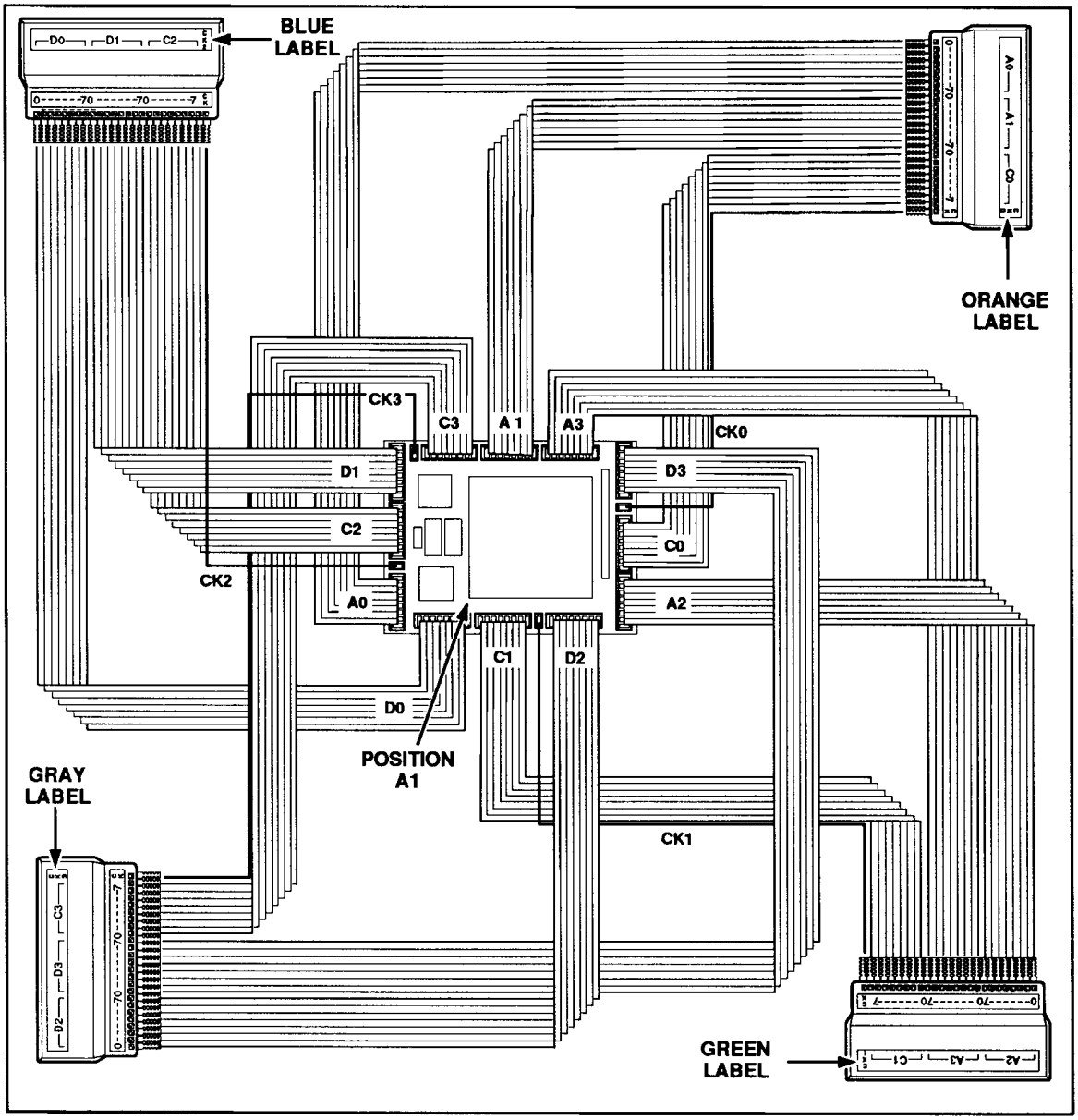


Figure 2-7. Connections from the 92A96 probe cables to the probe adapter.

Placing the Probe Adapter in the SUT

To place the probe adapter into the SUT, follow these steps:

1. Turn off the power to the SUT. (It is not necessary to power down the DAS 9200.)
2. Refer to Figure 2-8 and carefully plug the probe adapter into the R4000 system socket. Align position A1 on the probe adapter to position A1 on the R4000 system socket.

NOTE

It may be necessary to stack one or more replacement sockets between the probe adapter and SUT. This will gain vertical clearance from adjacent components. However, this may increase loading, which may reduce the electrical performance of the probe adapter.

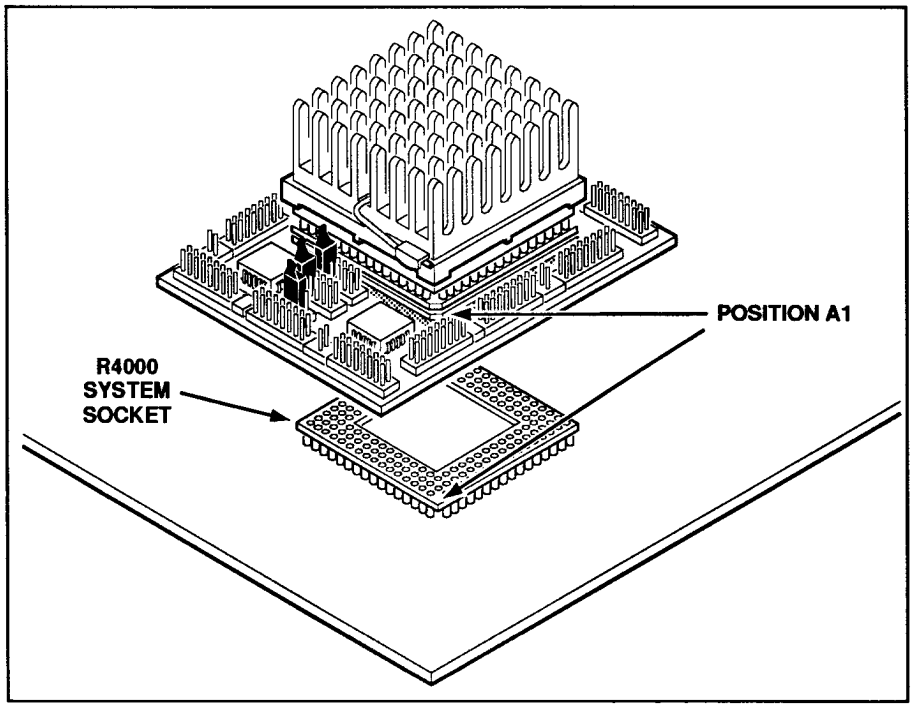


Figure 2-8. Placing the probe adapter into the R4000 system. The clock and 8-channel probes should already be connected before placing the probe adapter in the system. They are not shown connected in this figure because they would obscure the detail.

CAUTION

The R4000 microprocessor can be permanently damaged if you insert the probe adapter with the wrong orientation in the system socket. The probe adapter has an arrow that shows how to properly orient position A1 when installing it in the system socket. Figure 2-8 shows the proper alignment of position A1 on the R4000 system socket and the probe adapter socket.

Connecting the Interface Housings

The probe cables and interface housings may already be connected. If they are not connected, refer to Figure 2-9 and follow this procedure:

1. Select an interface housing with a label color that matches a label color on one of the 92A96 probe cables.
2. Line up the key on the loose connector end of the probe cable with the key slot on the interface housing and connect them.
3. Repeat steps 1 and 2 for each of the three remaining probe cables.

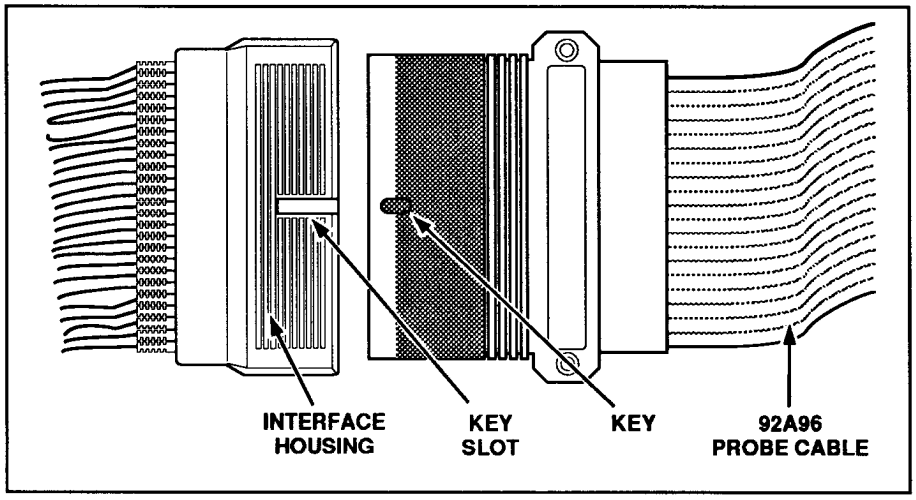


Figure 2-9. Connecting the interface housing to the 92A96 probe cable. The interface housing label is on the other side of the housing and not visible in the figure.

Section 3: ACQUIRING AND VIEWING DISASSEMBLED DATA

The primary function of this product is to assist you during the design period when you try to execute new or untested software on your prototype R4000 hardware. This section describes how to acquire data for viewing in the Disassembly menu.

After you install the microprocessor support software, make all the connections between the DAS 9200 and the SUT, and select the R4000 Support, you are ready to complete the setup for the data acquisition module. To complete the setup, you need to choose a clock type (or use the default) and define a trigger program (or use the default).

R4000 DISASSEMBLER OPERATION

There are times when the disassembly will be incorrect because the microprocessor does not indicate whether a read is filling the instruction cache or the data cache. Therefore, the disassembler cannot detect which cache is being filled. This means that all data reads and instruction fetches look alike to the disassembler.

The disassembler assumes that all noncoherent block reads for which you have specified the block size are instructions and will be disassembled. The disassembler also assumes that single-word reads that are word aligned and fall within the address range you have specified are instructions and will be disassembled. Refer to the description of *Instruction Cache Line Size* later in this manual for information about setting the I-Cache Line Size field of the Disassembly format Definition overlay to match the size of the Instruction Cache line used in your R4000 system.

Coherent block reads will normally be for data. However, when a program's code space and data space share the same "page," it is possible that instruction fetches can be with coherent block reads. In such cases, you can use the Mark Data function key to correct the disassembly. Non-coherent block reads will normally be for instructions but you may need to correct the disassembler similarly by using the Mark Data key. Refer to *Marking Cycles* in this section for information on how to change an erroneously disassembled instruction.

Read-with-Write-Forthcoming requests are treated the same as all other block read requests. The disassembler ignores whether or not the Link Address is retained, and whether or not the Exclusive bit is set.

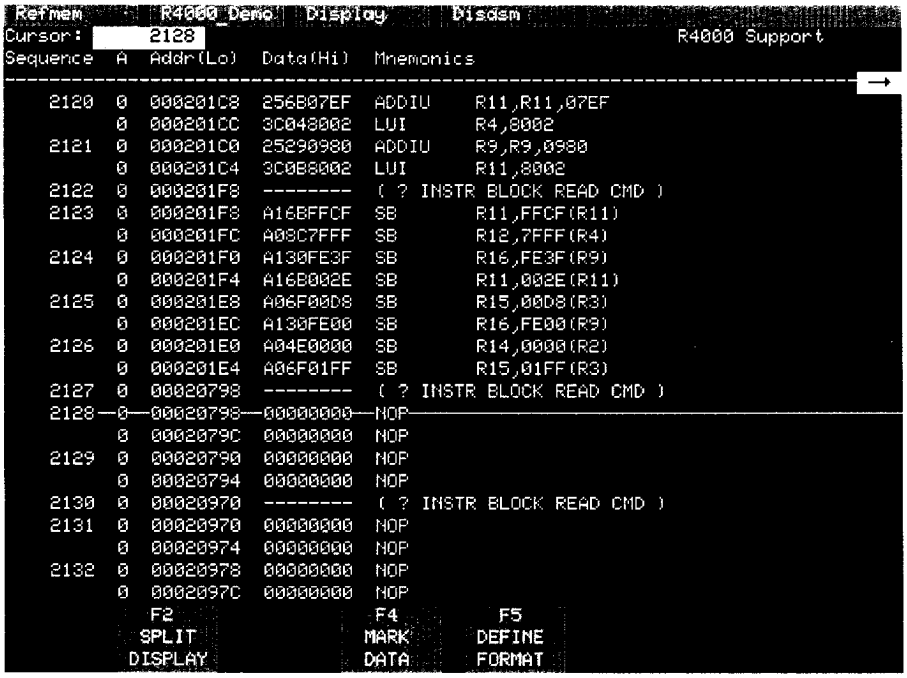


Figure 3-1. Example of a misinterpreted data read. Sequences 2128 and 2129 show a block of NOP instructions. Since non-coherent block reads may be for either data or instructions, the disassembler had assumed this block read was for instructions. However, it is more likely to be a block of data reads.

You can manually change the erroneously disassembled bus cycle using the F4: MARK DATA key of the Disassembly menu. Figure 3-10 shows this example after the NOP is corrected to a data read. Refer to the description on *Manually Overriding Disassembled Instructions* later in this section for directions on how to do this.

CLOCKING

You can use the Clock menu to set clocking choices to control data sampling. The 92DM75A software offers a customized clocking selection for the R4000 microprocessor. This clocking choice (Custom) is the default selection whenever you select R4000 Support in the Configuration menu.

Custom clocking only stores one data sample for each bus transaction, which can take one or more clock cycles.

A description of how cycles are sampled is found in Appendix B.

Disassembly will not be correct with the Internal or External clocking modes. Refer to *Section 4: Hardware Analysis* for a description of using these other clock selections with this microprocessor support package.

TRIGGERING

All the Trigger menu selections currently available for your data acquisition module are still valid for disassembly. Refer to your acquisition module user manual for a list and description of the selections.

When specifying an address in a trigger program, the address must be the physical address, not the virtual address. At the same time, you must also specify a command symbol, not a data symbol, for the Control group. If your microprocessor is using Sub-Block Ordering (see the microprocessor user manual), it may be difficult to trigger on the beginning address of any given block.

The symbol table for the Control group (R4000_Ctrl) contains over 300 symbols. To quickly clear the field for the Control group in the Trigger menu, open the field, press the Home key, and close the field. The first entry in the list is blank.

When acquiring data using Custom clocking, some signals should not be used as Channel events in the Trigger menu. If they are used, acquired data might be invalid. These signals are:

- ValidOut*= (You should use ValidOut* instead)
- RdRdy*_L
- WrRdy*_L

The DAS 9200 makes it possible to cross-trigger with other modules or to an external instrument. You may want to consider sending or receiving a signal to or from another DAS 9200 module, or to the Sync Out SMB connector on the module. You should refer to your *DAS 9200 System User Manual* for a description of defining and using signals; refer to your acquisition module user manual for a description of using the Sync Out SMB connector.

ACQUIRING DATA

After you set up the disassembler to acquire data from your R4000 system, you can press F1: START to begin the acquisition. After satisfying the trigger program and filling acquisition memory, the DAS 9200 displays data in the format used last. You may need to press F1: STOP if the trigger or stop conditions are not met. You can change the display menus from the Menu Selection overlay (press the Select Menu key).

If the trigger does not occur within a few moments, the DAS 9200 displays the Monitor menu showing the progress of the acquisition. You might want to check the trigger program when this occurs. Also refer to *Appendix A: Error Messages and Disassembly Problems* for a description of typical problems and possible solutions.

DISPLAYING DISASSEMBLED DATA

The DAS 9200 displays disassembled data in the Disassembly menu. This menu shows the disassembled bus cycles, instruction mnemonics, operands and addresses, and data values.

You can display the disassembled data in different formats, and you can select the disassembly display format and tailor it for your application using the Disassembly Format Definition overlay. Detailed information on this overlay is provided later in this section.

The disassembler software needs both a data cycle and a corresponding address cycle to disassemble data correctly. Because of this, you should be careful when qualifying data in the Trigger menu.

Display Formats

The R4000 disassembler software provides four formats for displaying disassembled data:

- Hardware format shows all acquired cycle types and instruction mnemonics in the order they occurred
- Software format suppresses all request cycles, all data read and write cycles, and displays a menu that looks similar to an assembly language program listing
- Control Flow format only displays the instructions that change the control flow of the microprocessor
- Subroutine format only displays subroutine calls, exceptions, and returns

You can further define how the data is displayed within these four formats by selecting various display options in the Disassembly Format Definition overlay.

Figure 3-2 shows an example of the Disassembly menu.

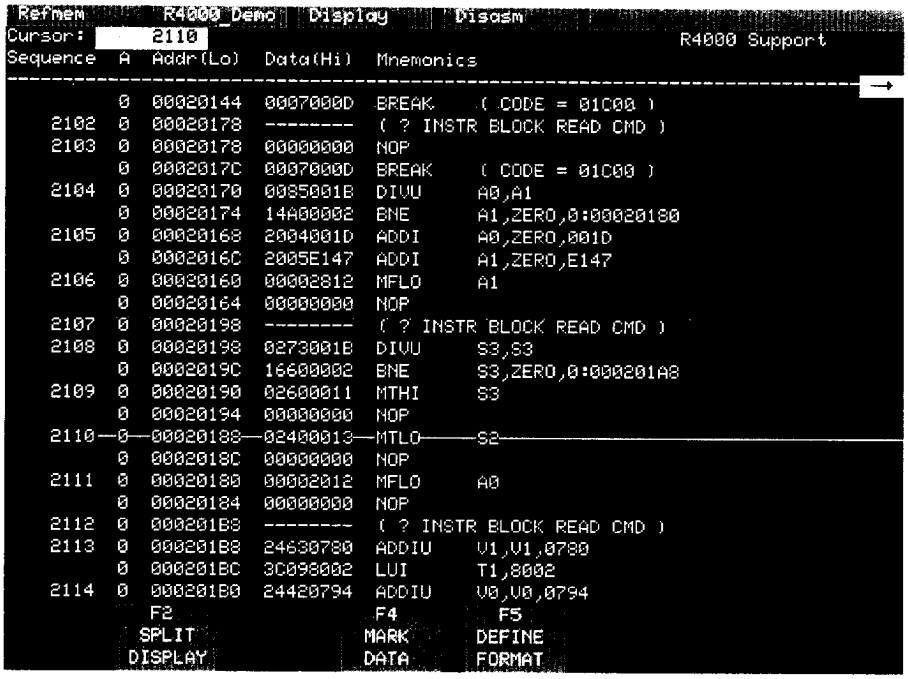


Figure 3-2. Disassembly menu. Software register names are shown for the CPU general registers.

No matter which display format you decide to use, the disassembler software displays information in the Disassembly menu in the following columns:

- **Sequence Column.** The sequence column shows the sequence number of the data displayed on that line. The cursor field in the upper-left area of the menu displays the sequence number of the current cursor location.

- **A Group Column.** The disassembler calculates the high-order 4 bits of the address and displays that value for the A group. If your R4000 system is restricted to a 32-bit physical address, you can make this group invisible in the Disassembly Format Definition overlay. However, making this group invisible will not stop the disassembler from displaying the appropriate value in the operand.

- **Addr(Lo) Group Column.** The Addr(Lo) group column shows values for the low-order 32 bits of the address at each sequence. The probe adapter determines the value during command cycles, and, based on that value, the disassembler synthesizes the addresses for the data cycles. In all other display and setup menus, these are the SysAD31-0 channels.

  You can display the Addr(Lo) group either symbolically or as an eight-digit hexadecimal value. To display the Addr(Lo) group symbolically, refer to *Displaying the Addr(Lo) Group Symbolically* in this section.

  For block reads and writes, the calculated address follows the rules of Sub-Block Ordering. For each double-word cycle, the address calculated for the first line will be modulo-8 and the address calculated for the second line will be modulo-8 plus 4.

  Gaps in the acquired data, caused by data qualification specified in the Trigger menu, are indicated by a gray background behind the Addr(Lo) group.

- **Data(Hi) Group Column.** The disassembler displays the value of the 32-bit wide data. Each double word (64-bit wide) cycle requires two lines. The first line displays the lower addressed word and the second line displays the higher addressed word. The disassembler displays two lines when valid data straddles a word boundary.

  The disassembler displays only one line when the valid data is completely contained in either half of the bus.

  In all other display and setup menus, these are the SysAD63-32 channels. During command cycles, this group will be displayed as dashes. You can display the Data(Hi) group as a hexadecimal value, an octal value, a binary value, or not display it at all.

  Gaps in the acquired data, caused by data qualification specified in the Trigger menu, are indicated by a gray background behind the Data(Hi) group.

- **Instruction Mnemonics Group Column.** The instruction mnemonics column shows the disassembled R4000 cycles and instructions. The disassembler displays disassembled instructions and operands as they are described in the *R4000 Microprocessor User's Manual* (1991).

  The disassembler displays all numeric operands as hexadecimal values. Register numbers are in decimal. When you select the Symbolic radix for the Addr(Lo) group, the calculated effective addresses are displayed symbolically.

PC-relative branch instructions will be displayed with absolute addresses calculated from the PC's current address. Since the PC's current address as seen by the disassembler is a physical address, the calculated absolute branch address will also be a physical address. As a result, the displayed branch address will normally not match the "virtual" address specified by the software. In addition, the user must be aware that incorrect physical branch addresses may be calculated and displayed in situations where program execution crosses Translation Lookaside Buffer page boundaries and the virtual-to-physical mapping changes.

For more information about how the address is calculated, refer to the description of the Translation Lookaside Buffer and PC-with-Displacement Calculations under *R4000 System Requirements and Restrictions* in Section 1.

One or two lines per cycle will be displayed for instruction, read, and write cycles, depending on the cycle type. The disassembler displays all block instruction, read, and write cycles as two lines per cycle. All double, single, and partial word read and write cycles are displayed as two lines per cycle when the number of bytes straddle the word boundary. Otherwise, they are displayed as one single line per cycle.

All single word (non-cachable) instruction cycles are displayed as only one line per cycle. The R4000 microprocessor does not have double word (non-cachable) instruction cycles.

- **Timestamp Column.** The timestamp column shows the timestamp value, when you choose to display the timestamp values. You can use the Timestamp field of the Disassembly Format Definition overlay to select Absolute, Relative, Delta, or Off.

  Timestamp values show the amount of time that has elapsed between data samples. An Absolute timestamp shows the amount of time elapsed between when the acquisition was started (after pressing F1: START) and each subsequent sample. A Relative timestamp shows the amount of time elapsed between successive samples. A Delta timestamp shows the amount of time between the sample with the delta user mark, and each previous and subsequent sample. (To place a delta mark on a sample, use F4: MARK DATA from the Disassembly menu.)

Cycles are assigned the value of the timestamp counter at the time that the 92A96 master clock occurs. Because the timestamp counter changes in 10 ns increments, timestamps for data acquired from an R4000 system with a clock rate that is not synchronous with the timestamp counter will be slightly skewed. For example, data acquired from an R4000 system with a clock rate of 30 MHz (33 ns) would be assigned a relative timestamp either of 30 or 40 ns. This skewing is not cumulative.

Keep in mind the following characteristics of the Disassembly menu when viewing disassembled data:

- all numeric values are shown unsigned (non-negative numbers) except for Delta timestamp values
- all numeric values are justified from the least significant bit
- the timestamp value is always displayed as a decimal value
- the timestamp for all the modules in a cluster is shown only in a single column
- the only radix selections for Mnemonics are ASCII or Off
- you can only select Symbolic radix when a symbol table is available for that group

The following discussions describe the four display formats.

Hardware Display Format

In the Hardware format, all bus cycles are shown in the order that they occurred. Instruction mnemonics are displayed on assumed instruction Fetch cycles and cycle-type information is displayed for all other cycles. The disassembler cannot detect flushes following a branch instruction. Non-instruction bus cycles are displayed as either 32-bit or 64-bit wide data transfers.

The disassembler shows only valid bytes on the data bus during data transfers depending on the selection made in the Byte Order field of the Disassembly Format Definition overlay. Byte order also determines which half of the SysAD bus gets decoded as instructions. Refer to the discussion on *Byte Ordering* later in this section. Invalid data bytes are represented by dashes.

Figure 3-3 shows the Hardware display format with hardware names displayed for the CPU general registers.

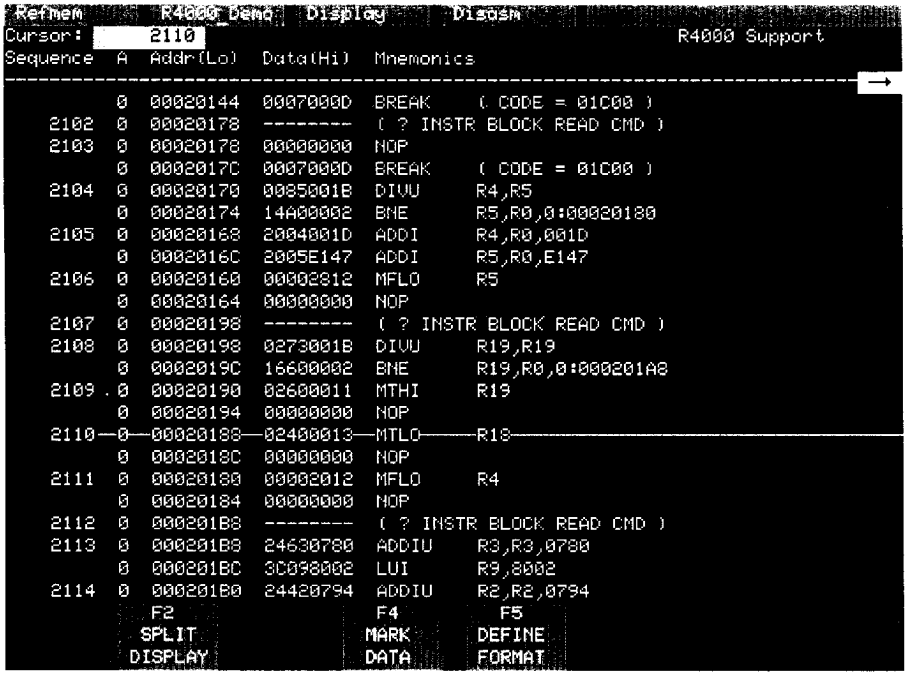


Figure 3-3. Hardware display format. Hardware register names are shown for the CPU general registers.

Software Display Format

In the Software format, all assumed instruction fetches are displayed. Labels that indicate the beginning of exception handler routines are also displayed. All other cycle types are suppressed. The disassembler cannot detect flushes following a branch instruction.

Instructions are always displayed in ascending address order within cache fills. A cache fill is a series of cycles initiated by a Block Read request and terminated with a last read cycle. If your R4000 system uses Sub-Block Ordering, a cache fill will occur in which the necessary first double word is received last and proceed in a complex but predetermined order.

When using Sub-Block Ordering, the sequence numbers and timestamps are not reordered but the instructions within them are. This includes the A, Addr(Lo), Data(Hi), Mnemonics, and Control groups. Each sequence number will still have only two lines of disassembled instructions. Because of this reordering, the relationship between the Sequence/Timestamp, and instructions will not be the same as in Hardware format.

If your R4000 system is not using Sub-Block Ordering, then it is using Sequential Ordering. The disassembler synthesizes the correct physical address, regardless of whether Sub-Block Ordering or Sequential Ordering is being used. For details, refer to the *MIPS R4000 Microprocessor User's Manual.*

In tracing program execution, it's more difficult to determine if a particular branch or jump was taken, or what the destination address was following a branch or jump instruction when Sequential Ordering is used.

Figure 3-4 shows the Software display format with software names displayed for the CPU general registers. This figure also shows the addresses in correct order for an R4000 system using Sequential Ordering; the display would have been the same for an R4000 system using Sub-Block ordering.

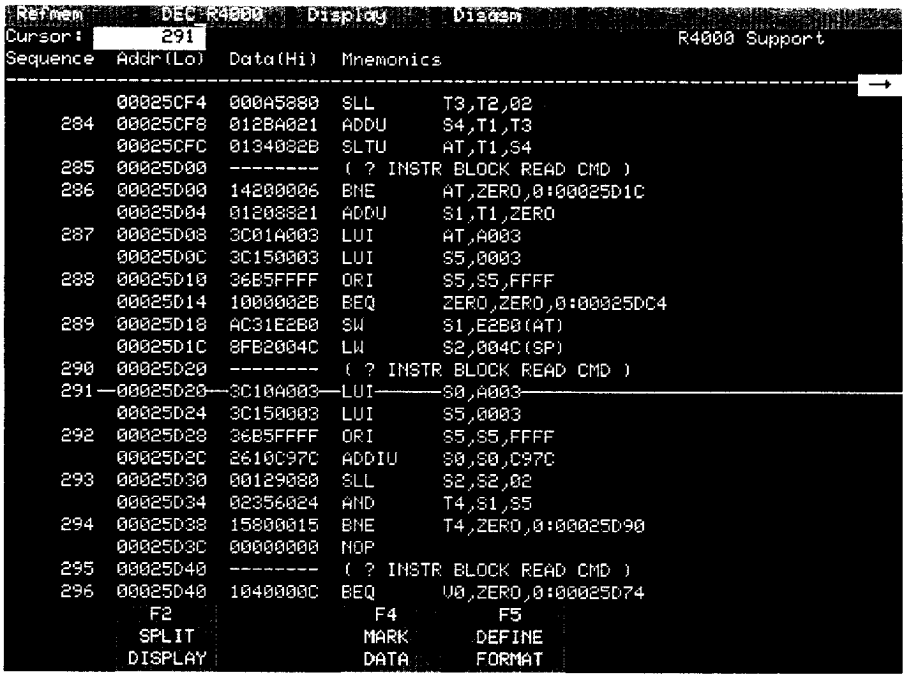


Figure 3-4. Software display format. Software register names are shown for the CPU general registers.

Control Flow Display Format

In the Control Flow format, only instructions that change the control flow are displayed. Some instructions that do not actually change the control flow are also displayed, such as a conditional branch that is not taken.

Control Flow instructions within cache fills are reordered in the same way that they are reordered in the Software format.

Exception handler entry labels and the instruction at that location will be displayed for control flow instructions. The label is always displayed regardless of the type of instruction.

Instructions that generate a change in the flow of control in an R4000 microprocessor are as follows:

BCzF	BGTZ	BREAK	TGE
BCzFL	BGTZL	ERET	TGEI
BCzT	BLEZ	J	TGEIU
BCzTL	BLEZL	JAL	TGEU
BEQ	BLTZ	JALR	TLT
BEQL	BLTZAL	JR	TLTI
BGEZ	BLTZALL	SYSCALL	TLTIU
BGEZL	BLTZL	TEQ	TLTU
BGEZAL	BNE	TEQI	TNE
BGEZALL	BNEL		TNEI

Figure 3-5 shows the Control Flow display format.

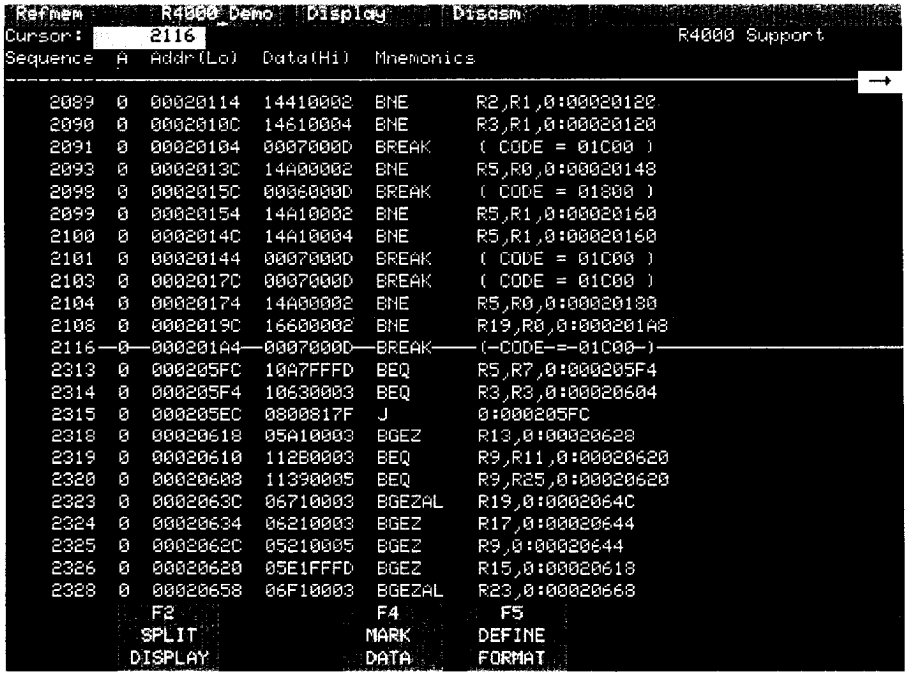


Figure 3-5. Control Flow display format.

Subroutine Display Format

The Subroutine format displays subroutine calls, exceptions, and returns only. Subroutine calls are assumed to be branch and jump instructions that perform a link. Because the disassembler cannot detect when a flush occurs, conditional branches are always displayed.

System Call, Break, and Trap on Condition instructions are also displayed. Exception handler labels as well as the instruction at that location will be displayed for subroutine instructions. The label is always displayed regardless of the type of instruction.

Subroutine instructions within cache fills are reordered in the same way that they are reordered in the Software format.

Instructions that appear in the subroutine display of a R4000 microprocessor are as follows:

BGEZAL	JAL	TGE	TLT
BGEZALL	JALR	TGEI	TLTI
BLTZAL	JR R31	TGEIU	TLTIU
BLTZALL	SYSCALL	TGEU	TLTU
BREAK	TEQ		TNE
ERET	TEQI		TNEI

Figure 3-6 shows the Subroutine display format.

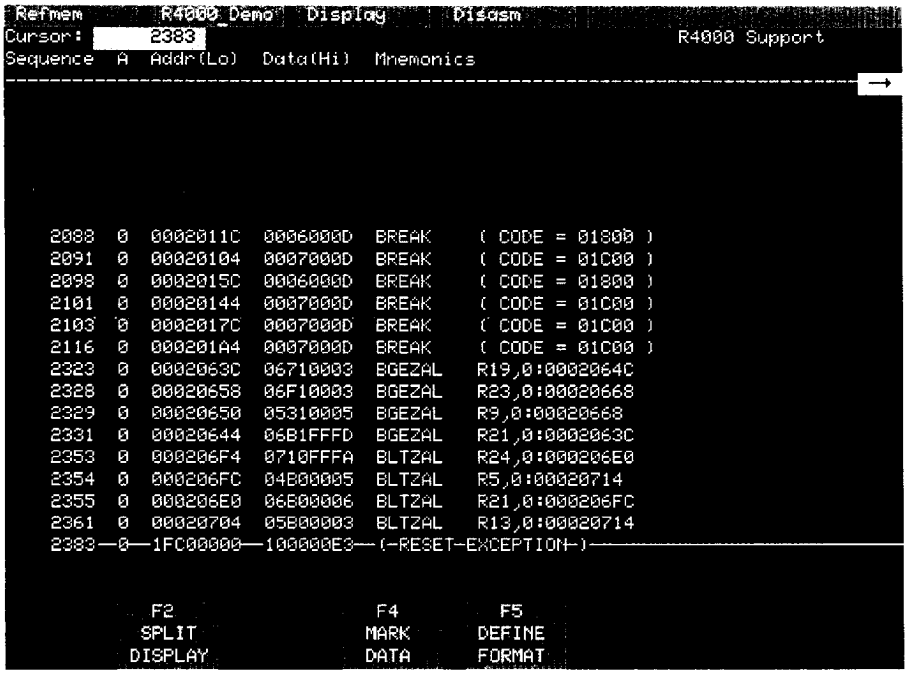


Figure 3-6. Subroutine display format.

Disassembly Format Definition Overlay

The Disassembly Format Definition overlay allows you to make optional display selections for the Disassembly menu and tailor it for your specific needs. To access this overlay, press F5: DEFINE FORMAT from the Disassembly menu.

You can use this overlay to do the following:

- choose the format (mode) in which the Disassembly menu displays disassembled data
- set the interval in which the data cursor will scroll through disassembled data
- display and define the format of the timestamp
- highlight various types of disassembled cycles or gaps
- continue disassembly across gaps in the acquisition
- choose software or hardware names for the CPU general registers
- choose Big- or Little-Endian byte ordering
- select the size of the instruction cache line
- enter the beginning address of the uncached area
- enter the size of the uncached area
- change the order in which the channel groups are displayed in the Disassembly menu
- change the radix for each group
- choose which symbol table to use for each group where symbolic is the selected radix

Figure 3-7 shows the Disassembly Format Definition overlay.

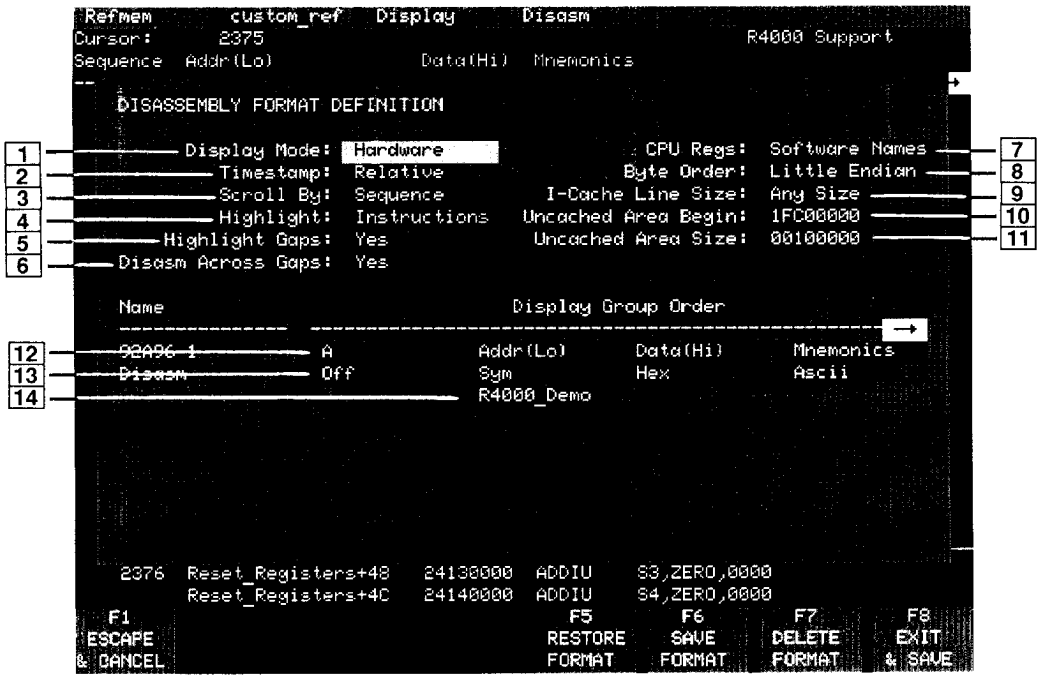


Figure 3-7. Disassembly Format Definition overlay.

1 **Display Mode.** You can display the disassembled cycle types or instruction mnemonics in Hardware, Software, Control Flow, or Subroutine modes.

2 **Timestamp.** You can display the timestamp as an Absolute, Relative, or Delta value. You can also set the timestamp display to Off. Refer to the description of timestamp column earlier in this section for definitions of these selections.

3 **Scroll By.** You can scroll by Sequence, Instructions, Control Flow, or Subroutines.

4 **Highlight.** You can highlight All, Instructions, Control Flow, or Subroutines. Only the selected type of samples are shown as white text with a black background with highlighting on. All other samples are shown as gray text with a black background.

5 **Highlight Gaps.** You can choose to highlight or not to highlight gaps. Gaps are caused by qualifying data storage in the Trigger menu and are indicated by a gray background behind the address and data values.

[6] **Disasm Across Gaps.** You can choose to continue or to discontinue to disassemble instructions across gaps. Disassembling instructions across gaps causes the disassembler to align the last address or data sample before the gap with the address or data sample immediately following the gap. Disassembled data will be invalid if these samples do not logically match.

[7] **CPU Regs.** You can select the names of the CPU general registers as either Software or Hardware.

[8] **Byte Order.** You can select Big- or Little-Endian byte ordering for data reads and writes.

[9] **I-Cache Line Size.** You can select the size of the instruction cache line as either Any Size, 4-word, 8-word, 16-word, or 32-word.

[10] **Uncached Area Begin.** You can enter the address of the beginning of the non-cached area. To generate a valid non-cachable address range, the beginning address and size of the non-cached area must not exceed a 32-bit integer.

[11] **Uncached Area Size.** You can enter the size of the non-cached area.

[12] **Group Name.** You can specify the name of the group that displays in the column in which the cursor is positioned. When you move a group, the group is inserted in the new column position and removed from its old position. All the groups to the right of the inserted group are moved over one column position to the right.

[13] **Group Radix.** You can select the radix in which each group displays. The radix selections for most groups are Binary, Octal, Hexadecimal, Symbolic, and Off. The only selections for Mnemonics are ASCII or Off. You should only select the Symbolic radix when a symbol table is available for that group. The timestamp value always displays in decimal.

[14] **Symbol Table.** You can specify a symbol table to use for each group where Symbolic is the selected radix.

Function Keys

F1: ESCAPE & CANCEL. Closes the overlay and discards any changes you have made since entering it.

F5: RESTORE FORMAT. Displays a list of saved disassembly formats. Use the cursor keys to select the desired format to restore and press the Open/Close or Return key.

F6: SAVE FORMAT. Saves the current selections for the Disassembly Format Definition overlay in a file on disk. You can enter a file name up to ten characters long.

F7: DELETE FORMAT. Displays a list of saved disassembly format files for the current module or cluster setup. Use the cursor keys to select the desired format to delete and press the Open/Close key. You cannot delete the Default format.

F8: EXIT & SAVE. Exits the overlay and executes or saves any changes made.

Register Names

You can select the type of names in the display for the R4000's CPU general registers by making a selection in the CPU Regs field of the Disassembly Format Definition overlay. The choices are Software or Hardware names. Table 3-1 shows the register names and their use for the R4000 microprocessor.

Table 3-1
CPU General Register Names and Usage

Hardware Name	Software Name	Usage
R0	ZERO	Value of 0
R1	AT	Assembler Temporary
R2 - R3	V0 - V1	Variables
R4 - R7	A0 - A3	Arguments
R8 - R15	T0 - T7	Temporary Registers
R16 - R23	S0 - S7	Saved Registers
R24 - R25	T8 - T9	Temporary Registers
R26 - R27	K0 - K1	Kernel Register
R28	GP	Global Pointer
R29	SP	Stack Pointer
R30	FP	Frame Pointer
R31	RA	Return Address

For opcodes that use the Floating Point coprocessor, the general register names are F0 through F31. All control registers have the manufacturer's software names. The disassembler will not flag improper or invalid register usage.

Bus Cycle Types

Cycle type labels are enclosed in parenthesis when the bus cycle is not a disassembled instruction. Table 3-2 describes the bus cycle types that the disassembler displays when viewing data in the Hardware format.

Table 3-2
Bus Cycle Types

Cycle Type	Description
Command Cycle Labels*	
? INSTR BLOCK READ CMD	These cycles are noncoherent block read requests. If they are of the user-specified block size, the following EXT RESP cycles will be disassembled. If not, they are assumed to be data.
? INSTR BLOCK RwWF CMD	These cycles are noncoherent block read-with-write-forthcoming requests. If they are of the user-specified block size, the following EXT RESP cycles will be disassembled. If not, they are asumed to be data.
PROC COHERENT BLOCK READ CMD	A processor command that causes a conherent block read request. The following EXT RESP cycles are assumed to be data.
PROC COHERENT BLOCK RwWF CMD	A processor command that causes a conherent block read-with-write-forthcoming request. The following EXT RESP cycles are assumed to be data.
PROC READ CMD: X BYTES	A single cycle read request by the processor. X is replaced with the number of bytes read (1 through 8). If they are for 4-bytes that are word aligned, and fell within the user-specified noncachable address range, they are assumed to be commands requesting noncacheable instructions, and the following EXT RESP cycle will be disassembled. Otherwise, they are assumed to b e data.
PROC WRITE CMD: X BYTES	A single cycle write request by the processor. X is replaced with the number of bytes written (1 through 8).
PROC BLOCK WRITE CMD	A request indicating the processor will write a cache line to memory.
PROC NULL WRITE CMD	A request indicating the processor will not write to memory. these can only occur following a PROC RwWF request, and there will be no data cycle following this request.
PROC INVALIDATE CMD	A request indicating that the memory or other cache value for the given address has been superseded by the processor cache. processor invalidates must be acknowledged or canceled. The following data cycle is unused.
PROC UPDATE COMP CMD: X BYTES	A request to update the value of the given address in other caches. Compulsory updates must be acknowledged or canceled. X is replaced with the number of bytes updated (1 through 8).

(Cont.)

Table 3-2 (Cont.)
Bus Cycle Types

Cycle Type	Description
PROC UPDATE POT CMD: X BYTES	A request to update the value of the given address in other caches. Potential updates may or may not be acknowledged or canceled. X is replaced with the number of bytes updated (1 through 8).
EXT READ CMD: X BYTES	An external system request that causes a read from the processor. X is replaced with the number of bytes read (1 through 8). A bus error will result for this request from the R4000 processor.
EXT WRITE CMD: X BYTES	An external system request that initiates a write to the processor. X is replaced with the number of bytes written (1 through 8). This request is only useful to set interrupts in the R4000 microprocessor.
EXT SYSTEM INTERFACE REL CMD	A null request that returns the system interface bus to master state. No data cycles follow this request.
EXT SECONDARY CACHE REL CMD	A null request that returns ownership of the secondary cache to the microprocesor. No data cycles follow this request.
EXT INVALIDATE CANCEL CMD	The external system is invalidating a line of the processor cache. This invalidate conflicts with a previous invalidate or update issued by the processor (but not acknowledged), so this request cancels that previous request. The following data cycle is unused.
EXT INVALIDATE CMD	The external system is invalidating a line of the processor cache. The following data cyce is unused.
EXT UPDATE CANCEL CMD: X BYTES	The external system is updating the processor cache. Up to a double word may be updated (more than a double word requires multiple update requests). This update is in conflict with a previous invalidate or update issued by the processor (but not acknowledged), so this request cancels that previous request. X is replaced with the number of bytes updated (1 through 8).
EXT UPDATE CMD: X BYTES	The external system is updating the processor cache. Up to a double word may be updated (more than a double word requires multiple update requests). X is replaced with the number of bytes updated (1 through 8).
EXT INTERVENTION CANCEL CMD	This is a request to the processor to return the status of a specific cache line, and sometimes the contents of that cache line. This intervention conflicts with a previous invalidate or update issued by the processor (but not acknowledged), so this request cancels that previous request.
EXT INTERVENTION CMD	This is a request to the processor to return the status of a specific cache line, and sometimes the contents of that cache line.

(Cont.)

Table 3-2 (Cont.)
Bus Cycle Types

Cycle Type	Description
EXT SNOOP CANCEL CMD	This is a request to the processor to return the status of a specific cache line. This snoop conflicts with a previous invalidate or update issued by the processor (but not acknowledged), so this request cancels that previous request.
EXT SNOOP CMD	This is a request to the processor to return the status of a specific cache line.
Data Cycle Labels*	
DATA ERROR	A report of a data error by either the processor on the external system.
EXT RESP DATA: (cache state)	Data returned by the external system in response to a request by the processor. If this data is in response to a PROC COHERENT BLOCK READ request or PROC CONHERENT BLOCK RwWF request, the current cache state will be appended.
EXT RESP LAST DATA: (cache state)	Data returned by the external system in response to a request by the processor. This is the last cycle satisfying this request. If this data is in response to a PROC COHERENT BLOCK READ request or PROC CONHERENT BLOCK RwWF request, the current cache state will be appended.
EXT WRITE DATA	Data provided by the external system after an external write request.
EXT WRITE LAST DATA	Data provided by the external system after an external write request. This is the last cycle of this data.
EXT UPDATE LAST DATA	Data provided by the external system after an external update request. This is the last cycle of this data.
PROC RESP DATA: (cache state)	Data provided by the processor in response to a request by the external system. If this data is in response to an EXT INTERVENTION request or EXT SNOOP request, the current cache state will be appended.
PROC RESP LAST DATA: (cache state)	Data provided by the processor in response to a request by the external system. This is the last cycle satisfying that request. If this data is in response to an EXT INTERVENTION request or EXT SNOOP request, the current cache state will be appended.
PROC WRITE DATA	Data provided by the processor after a processor write request.
PROC WRITE LAST DATA	Data provided by the processor after a processor write request. This is the last cycle of this data.
PROC UPDATE LAST DATA	Data provided by the processor after a processor update request. This is the last cycle of this data.

(Cont.)

Table 3-2 (Cont.)
Bus Cycle Types

Cycle Type	Description
UNUSED DATA CYCLE	An unused data cycle following a PROC INVALIDATE request or EXT INVALIDATE request.
(cache state)	Cache state information is appended to an EXT RESP when it is in response to a PROC COHERENT BLOCK READ request, or PROC COHERENT BLOCK RwWF request and is appended to a PROC RESP when it is in reponse to an EXT INTERVENTION request or EXT SNOOP request. Cache states can be: INVALID CLEAN EXCLUSIVE DIRTY EXCLUSIVE SHARED DIRTY SHARED
Special Cycle Labels	
BUS FAULT	A cycle in which both ValidIn* and ValidOut* are asserted, making it an invalidate cycle type. This will not occur with a properly operating system.
BUS IDLE	A cycle in which neither ValidIn* nor ValidOut* are asserted. Even though this will normally occur, this cycle type is not acquired with Custom clocking.
UNKNOWN	A control group bit pattern that does not match any defined system bus transaction. This will not occur with a properly operating system.
ILLEGAL INSTRUCTION	CPU instructions in the R4000 Opcode Bit Encoding table in the *Mips R4000 Microprocessor User's Manual* that causes a reserved instruction exception or are invalid but do not cause a reserved instruction exception.
ILLEGAL COP0 INSTRUCTION	CP0 instructions, classified as γ (gamma) in the *Mips R4000 Microprocessor User's Manual*, that cause a reserved instruction exception.
ILLEGAL COP0 FUNCTION	CP0 functions, classified as ϕ (phi) in the *Mips R4000 Microprocessor User's Manual*, that do not cause a reserved instruction exception.
ILLEGAL COP1 INSTRUCTION	CP1 instructions, classified as γ (gamma) in the *Mips R4000 Microprocessor User's Manual*, that cause a reserved instruction exception.
ILLEGAL COP1 FUNCTION	CP1 functions, classified as δ (delta) in the *Mips R4000 Microprocessor User's Manual*, that cause an unimplemented operation exception.
RESET	A reset cycle.

*Displayed in Hardware format only.

Figure 3-8 shows how some of these cycles are displayed by the DAS 9200. Refer to *Appendix B: How Data is Acquired* to see when the various signals used to disassemble data are sampled.

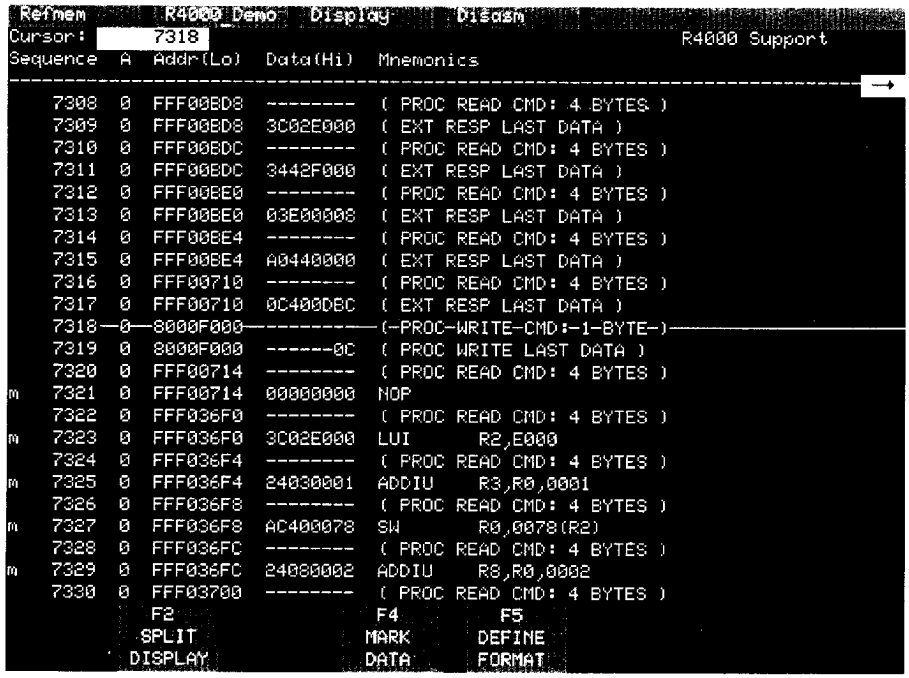


Figure 3-8. Displayed cycle types.

R4000 Exception Labels

The R4000 microprocessor does not perform any special bus cycles during exception processing such as vector fetch. Instead, the CPU saves the current program counter value in a special internal register and begins execution at a small set of fixed exception handler entry points. All software exceptions and most hardware exceptions have the same entry point; the exception handler code sorts out the various exceptions.

The disassembler places a special label immediately before any instruction fetched from one of the exception label points as shown in Table 3-3. The upper 4 bits (ninth digit) of the address is always assumed to be 0.

Table 3-3
Exception Labels

Address	Label
0x1FC0 0000	RESET EXCEPTION
0x1FC0 0200	TLB REFILL EXCEPTION
0x1FC0 0280	XTLB REFILL EXCEPTION
0x1FC0 0300	CACHE ERROR EXCEPTION
0x1FC0 0380	COMMON EXCEPTION

The disassembler entry points assume that BEV=1. Because of this, the exception entry points are noncachable. In normal usage, the exception entry points contain a Branch or Jump instruction, and therefore serve as an exception vector. However, the instructions at each of these locations is displayed according to selections in the Disassembly Format Definition overlay.

The exception entry points are independent of the Noncachable Address Range you specify. If you have changed the address range to not encompass the addresses of the entry points, then these instructions will not be disassembled; only labels will display.

Byte Ordering

You can select either Big- or Little-Endian byte ordering for data reads and writes in the Byte Order field of the Disassembly Format Definition overlay. Big-Endian byte ordering is when the most significant data byte of a multibyte number is located at the lowest address. Little-Endian byte ordering is when the least significant data byte of a multibyte number is located at the lowest address. The byte ordering must match between the R4000 system and the disassembler for correct disassembly.

Only Valid Bytes Displayed

The disassembler displays only the valid bytes of data for double, single, and partial word reads and writes. Invalid bytes, half-words, and words display as dashes. The valid bytes are determined by the type of transfer, the address of the transfer, and the selected byte order. Dashing will only occur when the radix for the Data(Hi) group is hexadecimal.

When the number of bytes does not cross a word boundary, only the affected word is displayed, with the appropriate bytes displayed as dashes. Otherwise, both words are displayed with the appropriate bytes displayed as dashes.

Instruction Cache Line Size

You can select the size of the Instruction Cache line in the I-Cache Line Size field of the Disassembly Format Definition overlay. The choices are Any Size, 4-word, 8-word, 16-word, or 32-word. The default is Any Size which is the same as "don't cares." The default address range for the non-cachable instructions is 0x1FC00000 to 0x1FCFFFFF. The ninth digit is assumed to be 0 for all non-cachable instructions.

Refer to the description of *R4000 Disassembler Operation* earlier in this section for details on the display of instruction cache information.

Displaying the Addr(Lo) Group Symbolically

The Addr(Lo) group can be displayed as symbol values similar to the way the Control, Intr, and Ack groups can be displayed as symbol values. You can use the Symbol Editor menu to create symbol tables to assign symbols to specific addresses or various address ranges; then use the Channel menu to change the default radix of the Addr(Lo) group to SYM and assign to it the symbol table. All effective addresses are sent to the table before being displayed. Refer to your *DAS 9200 System User Manual* for a description of how to create symbol tables.

You can also change the radix of the Addr(Lo) group to SYM in the Disassembly Format Definition overlay. Figure 3-9 shows the Addr(Lo) group displayed symbolically. The symbol table file used for the Addr(Lo) group in this figure is the R4000_Demo symbol file. It is only valid for the R4000_Demo reference memory file.

```
Refmem      R4000_Demo   Display     Disasm
Cursor:       334                                    R4000 Support
Sequence  Addr(Lo)              Data(Hi)  Mnemonics
-------------------------------------------------------------------------→
          0002074C              01A00008  JR        R13
    324   00020740              00A62824  AND       R5,R5,R6
          00020744              3C0D8002  LUI       R13,8002
    325   00020778              --------  ( ? INSTR BLOCK READ CMD )
    326   00020778              3C06BFC0  LUI       R6,BFC0
          0002077C              00C00008  JR        R6
    327   00020770              00000000  NOP
          00020774              30C60000  ANDI      R6,R6,0000
    328   00020768              25EF0754  ADDIU     R15,R15,0754
          0002076C              01E00008  JR        R15
    329   00020760              24350002  ADDIU     R21,R1,0002
          00020764              3C0F8002  LUI       R15,8002
    330   RESET VECTOR          --------  ( PROC READ CMD: 4 BYTES )
    331   RESET VECTOR          100000E3  ( RESET EXCEPTION )
          RESET VECTOR          100000E3  BEQ       R0,R0,0:Int_Service_Routine
    332   1FC00004              --------  ( PROC READ CMD: 4 BYTES )
    333   1FC00004              00000000  NOP
    334—Int_Service_Routine———————(—PROC—READ—CMD:—4—BYTES—)————————————
    335   Int_Service_Routine   03E0D821  ADDU      R27,R31,R0
    336   1FC00394              --------  ( PROC READ CMD: 4 BYTES )
    337   1FC00394              401AF000  MFC0      R26,ERROREPC

          F2          F4          F5
        SPLIT       MARK       DEFINE
       DISPLAY      DATA       FORMAT
```

Figure 3-9. Addr(Lo) group displayed symbolically.

Marking Cycles

You can mark a data sample to easily identify data samples, to move quickly to a data sample, to manually correct a disassembled bus cycle, or to calculate delta timestamp measurements. You can mark a data sample with an A through M, a delta mark (Δ), or the correct bus cycle using the F4: MARK DATA feature of the Disassembly menu. An "m" is placed next to any sample that has had the bus cycle manually changed with an opcode mark.

An Undo Mark selection is also available for opcode marks. You can use both data and opcode marks on a data sample.

Manually Overriding Disassembled Instructions

Although the disassembler generally disassembles bus cycles correctly, there are situations where disassembly is incorrect. The disassembler makes the following assumptions:

- all noncoherent block reads that are of a user-specified cache line size are instructions
- all coherent block reads are data
- all 4-byte reads that are word aligned and fall within the user-specified Noncachable Address Range are instructions
- all other n-byte reads are data

Disassembly can be incorrect when the disassembler is not acquiring enough R4000 information to accurately disassemble the bus cycles. For example, this can occur when using storage qualification in the Trigger menu to qualify out request cycles.

Opcode marks are only available if the disassembler detects the Read or RwWF request associated with the data in the sample on which the cursor is positioned in the Disassembly menu. This means that the disassembler must be able to find the request associated with an External Response Data or External Response Last Data before any opcode mark selections display when you press the F4: MARK DATA key.

When a cycle is marked, the disassembler attempts to modify the disassembly of all cycles within that cache line fill (block read). If there are other marks in that cache fill line, the disassembler uses the mark on the lowest sequence number for that block and ignores the other marks.

To manually correct invalid disassembly, you can use the F4: MARK DATA key to change an erroneously disassembled bus cycle. Table 3-4 shows the selections and their functions when used to correct the invalid disassembly. The choices will vary (instructions/data) depending on the type of cycle the cursor is positioned on.

Table 3-4
Opcode Mark Selections

Opcode Mark	Function
Block:	
Fetch Block	The block transfer is interpreted as instruction fetches
Read Block	The block transfer is interpreted as data reads
Undo Mark	Removes the opcode mark from the sample on which the cursor is currently positioned
Word:	
Fetch	The word is interpreted as an instruction fetch
Read	The word is interpreted as a data read
Undo Mark	Removes the opcode mark from the sample on which the cursor is currently positioned

To correct a bus cycle, follow these steps:

1. Place the cursor on the data sample you want to change in the Disassembly menu.
2. Press F4: MARK DATA and select a bus cycle type.

   When you press F4: MARK DATA, a list of selections appears. The list will change from sample to sample to reflect logical selections for that individual sample. Unlike data marks that can only be used once, opcode marks can be used as often as necessary.

   If the data sample already has an opcode mark on it, then the selection list appears with the cursor on that mark. To remove the opcode mark from the sample on which the cursor is positioned in the Disassembly menu, select Undo Mark.
3. Press the Return key.

The data sample will reflect the corrected bus cycle and other samples may be affected according to the correction. An "m" is placed at the beginning of the data sample that has had the disassembled bus cycle manually changed. Figure 3-10 shows cycles corrected to data reads using F4: MARK DATA.

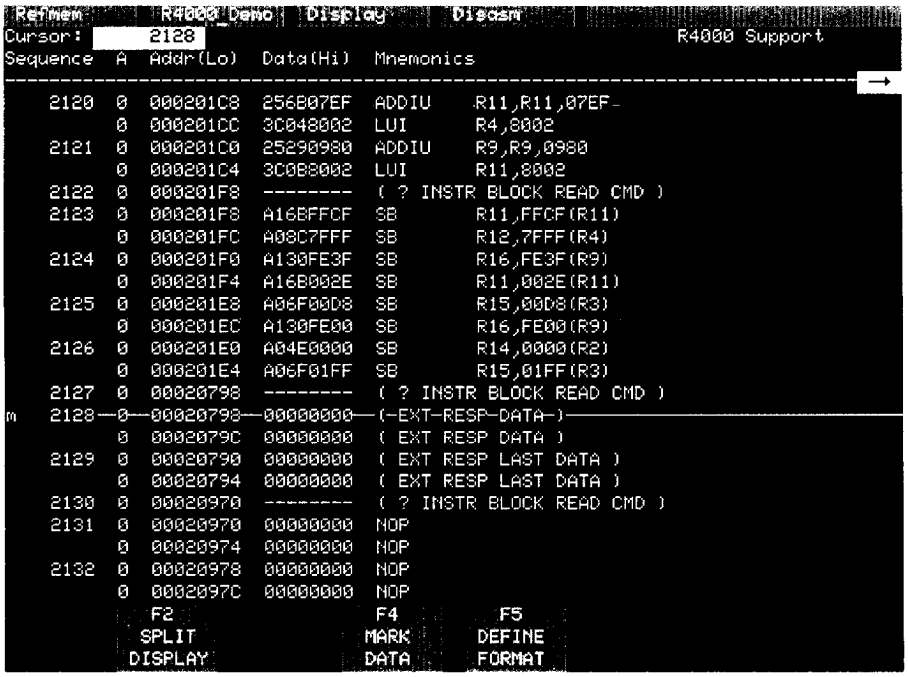


Figure 3-10. A corrected data read cycle after using F4: MARK DATA.

Marking a Data Sample

Marks can be used to make data samples easy to identify, and to quickly move the cursor to a marked sample. A special mark, the delta mark (Δ), is also used to calculate delta timestamp values.

The available marks are A through M and Δ. When you mark a data sample, that mark is attached to the acquired data. The mark will show with the data sample in all other display menus until you change it. If you are viewing the data in a split-screen display, such as when performing a search, the mark is attached only to the window in which it is placed and will not be carried over to the other window.

A small arrow appears at the beginning of the line of the marked data sample to make the mark more visible. The arrow disappears when both a data mark and an opcode mark are placed on the same sample.

To place a mark on a data sample, follow these steps:

1. Place the cursor on the data sample you want to mark in the Disassembly menu.
2. Press F4: MARK DATA and select a mark.

   When you press F4: MARK DATA, a list of selections appears. The cursor in the selection list will always be on the next unused mark. If you select a mark that has already been used, the previously marked sample will be unmarked and the current sample will be labeled with that mark when you close the selection list. Each mark can only be used on one data sample.
3. Press the Return key.

You can remove all the data marks in the Disassembly menu by using the State menu. To remove all data marks, follow these steps:

1. Press the Select Menu key to return to the Menu Selection overlay.
2. Select the State menu and press the Return key.
3. Press F5: DEFINE FORMAT.
4. Press F2: REMOVE MARKS and press the Return key. All the data marks will disappear.
5. Press the Select Menu key to return to the Menu Selection overlay.
6. Select the Disassembly menu and press the Return key. No data marks appear in the Disassembly menu; they have all been removed.

You can use the Δ mark to make delta timestamp measurements. After placing a Δ mark on a data sample, you can select the Delta selection in the Timestamp field of the Disassembly Format Definition overlay. When you press F8: EXIT & SAVE, the Disassembly menu samples will show the amount of elapsed time between the data sample with the Δ and each previous and subsequent sample.

To quickly move from one data mark to another, enter the mark for the new location in the Cursor field of the Disassembly menu. Enter an ^ in the Cursor field to move the cursor to a Δ mark.

Moving the Cursor

When displaying data in the Disassembly format, you can move the cursor directly to the sequence number or enter a specific sequence number in the Cursor field in the display menu and press the Return key.

Searching Through Data

The disassembler does not have a Search Definition overlay. However, you can effectively search through disassembled data by following these steps:

1. Press F2: SPLIT DISPLAY to use the Split-Screen overlay.
2. Select the Disassembly menu for one half of the split-screen display and the State menu for the other half.
3. Press F5: SPLIT HORIZ to split the screen into two horizontal displays.
4. Press F2: LOCK CURSORS. A selection list appears.
5. Select **lock cursors at the same sequence** and press the Return key.
6. Press F8: EXIT & SAVE to display the menus in a split screen.
7. If the active menu is the Disassembly menu, press F3: SWITCH WINDOW to make the State menu active. The active menu is the one with the yellow cursor and Cursor field.
8. Press F6: DEFINE SEARCH to use Search function of the State menu to search for the desired value.

   To abort a search, press the Esc (escape) key.

When searching for data in a clustered module setup in the State menu, the searches are conducted only for the master module. Refer to the description of the State Search Definition overlay in your module user manual for a description of how to search through state data. Also refer to that manual for a description of how to return to full screen display.

Figure 3-11 shows the screen split into Disassembly and State windows with the cursors locked on the same bus cycle.

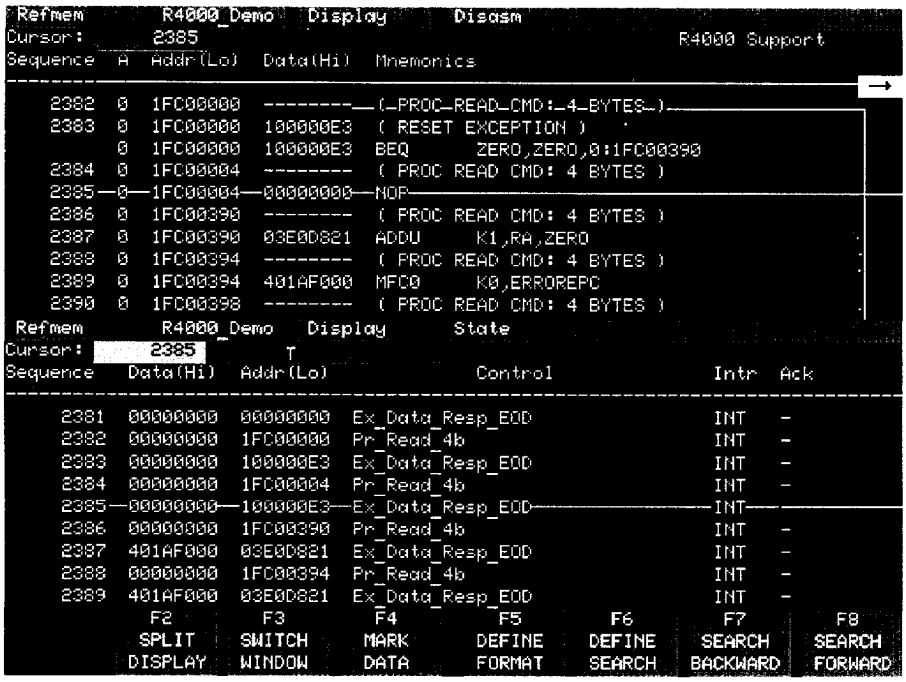


Figure 3-11. Disassembly and State split-screen display used to perform searches.

PRINTING DATA

To print disassembled data, you can use the Disassembly Print overlay. To access this overlay, press the Shift and Print keys at the same time from the Disassembly menu. The Disassembly Print overlay is exactly like the State Table Print overlay. Refer to your *DAS 9200 System User Manual* for a detailed description of the selections available in the State Table Print overlay.

Section 4: HARDWARE ANALYSIS

You may need to perform hardware analysis on your R4000 system prior to, during, and after attempting to integrate your software with the R4000 system hardware. When performing hardware analysis, you will probably want to use the data acquisition module to acquire data with a finer resolution. When more data samples are taken in a given period of time, the resolution in the Timing display increases, letting you see signal activity that would otherwise go undetected.

To acquire and display timing data, you need to change the clocking selection and trigger program, acquire data, and view it in the Timing menu. A predefined Timing Format Definition overlay file called R4000_96 (with bus forms) can be used to view R4000 timing data. A description of this file and how to use it can be found later in this section.

There often is a need to view data in a split screen display with state data in one half and timing data in the other. You should leave nearly all of the 92A96 probe cables or interface housings connected if you are analyzing data in this manner. Refer to Appendix B for information on the probe(s) you can disconnect.

CLOCKING

To change the data sampling rate, use the Clock menu.

The Normal/Timing jumper, J1411, on the probe adapter should be set in the Timing position to acquire timing data. For more information on the Normal/Timing jumper, refer to Section 2.

When using the 92A96 Module for timing analysis, you will want to use the Internal or External clocking modes. The Internal clock selection can sample data up to 100 MHz, which has a 10 ns resolution between samples. The External clock selection samples data on every active clock edge on the 92A96 clock inputs up to 100 MHz. If both clock edges are used (rising and falling), then the maximum clock rate is 50 MHz. Custom clocking for the R4000 microprocessor only uses the rising edge.

The default clocking mode is Custom when microprocessor support is used; you will need to change it to either Internal or External. Your data acquisition module user manual contains an in-depth description of the Clock menu, and the Internal and External selections.

Custom Clocking

Custom clocking only stores one data sample for each bus transaction, which can take one or more clock cycles. Custom clocking also time-aligns certain signals that otherwise would be skewed relative to the current bus transaction. Refer to Appendix B for an in-depth description of how Custom clocking acquires data.

Internal Clocking

When you select Internal as the clocking mode, the data acquisition module stores one data sample as often as every 10 ns (100 MHz). This clocking selection is commonly referred to as asynchronous.

Two typical uses of Internal clocking are to verify that all of the R4000 signals are transitioning as expected, or to measure the timing relationship between signal transitions.

External Clocking

When you select External as the clocking mode, the data acquisition module acquires and stores data based on the clock channel up to 100 MHz (using only one clock edge). If both clock edges are used (rising and falling), then the maximum clock rate is 50 MHz. This clocking selection is commonly referred to as synchronous.

The data acquisition module will sample data on every rising (or falling) edge of the selected clock, when you select the rising (or falling) edge of any clock available on the data acquisition module as the clock channel.

For External clocking of the R4000 system, you should select the rising edge of TClock0=. Both edges can also be used.

TRIGGERING

All the Trigger menu selections currently available for your data acquisition module are still valid for hardware analysis. Refer to your acquisition module user manual for a list and description of the selections.

ACQUIRING DATA

You can acquire data as described in the *Acquiring Data* description in Section 3. The Normal/Timing jumper, J1411, on the probe adapter should be set in the Timing position to acquire timing information.

The Addr(Lo) column in the Disassembly display shows values for the low-order part of the SysAD bus. In all other setup and display menus, these are the SysAD31-0 channels.

The Data(Hi) column in the Disassembly display shows values for the high-order part of the SysAD bus. In all other setup and display menus, these are the SysAD63-32 channels.

DISPLAYING DATA

Hardware analysis requires that you view data in either the Timing or State menus. It also requires that the Normal/Timing jumper be placed in the Timing position.

With the Normal/Timing jumper in the Timing position, the ValidIn*, ValidOut*, RdRdy*, and WrRdy* signals are delayed by 5 ns. When the Normal/Timing jumper is in the Normal position, ValidIn* and ValidOut* are delayed by 1 clock cycle, and RdRdy* and WrRdy* are delayed by 3 clock cycles when using Internal or External clocking.

Timing Menu

In the Timing menu, each individual channel is shown as a waveform, and groups of channels are shown as bus forms.

A predefined Timing Format Definition overlay file, part of the R4000 microprocessor support, is available for you to use when displaying data in the Timing menu. The R4000_96 file was installed on the DAS 9200 with the application software.

This timing format places the system clock as the first displayed channel followed by Data(Hi) and Addr(Lo) group signals (R4000 SysAD63-32 and SysAD31-0 signals) and other important control signals. The Data(Hi) and Addr(Lo) group signals are shown as bus forms containing bus values instead of individual timing waveforms. Figure 4-1 shows data displayed using the R4000_96 Timing Format file.

When viewing the Timing menu using the R4000_96 timing format file, also keep the following in mind:

- The most significant hexadecimal digit of the Control group can be decoded as follows:

  A stands for an External Command
  B stands for External Data
  C stands for a Processor Command
  D stands for Processor Data

- The Misc2 group contains the SysCmdP parity bit as its most significant bit and the SysAD7–SysAD0 parity bits as its least significant two hexadecimal digits.

To select the R4000_96 Timing Format file, follow these steps:

1. Select the Timing menu and press F5: MOVE TO DISPLAY or press the RETURN key.
2. Press F5: DEFINE FORMAT.
3. Press F5: RESTORE FORMAT.
4. Select R4000_96 and press the Return key. A message tells you the format file is selected.
5. Press F8: EXIT & SAVE to return to the Timing menu.

Refer to Table C-4 for the list of individual channels and their R4000 names.

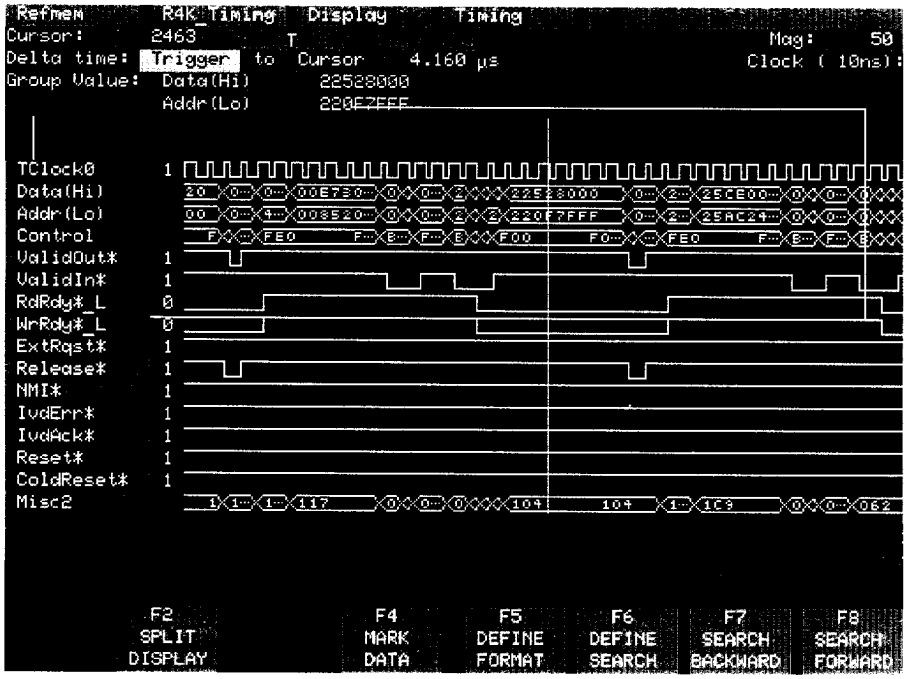


Figure 4-1. Timing data displayed using the R4000_96 Timing Format file. The clocking selection is Internal. Data acquired using External clocking would look similar except the TClock0 trace would be flat.

State Menu

In the State menu, all channel group values are shown based on the selected radix in the Channel menu or the State Format Definition overlay. This menu shows the bus cycles with address, data, and control values.

You may want to turn on the visibility for the Misc1 or Misc2 groups by accessing the State Format Definition overlay and changing their radix from Off to Hex, Bin, or Oct. You may also want to add the Timestamp field and change its radix as well.

VIEWING AND SEARCHING THROUGH DATA

You can view and search through State and Timing data as described in your acquisition module user manual.

Before performing a search in the Timing menu, be sure to check the State Format Definition overlay and make sure the channels on which you want to conduct a search are visible. Channels visible in the Timing menu can't be searched on unless they are visible in the State menu.

PRINTING DATA

To print state data, use the State Table Print overlay. To access this overlay, press the Shift and Print keys at the same time from the State menu.

To print timing data, use the Timing Print overlay. To access this overlay, press the Shift and Print keys at the same time from the Timing menu.

Refer to your acquisition module user manual for a description of these overlays.

Appendix A: ERROR MESSAGES AND DISASSEMBLY PROBLEMS

This section describes error messages and disassembly problems that you may encounter while acquiring data.

MODULE ERROR MESSAGES

These error messages will appear in the Module Monitor menu when there are problems with acquiring data or satisfying the trigger program. The error messages are listed in alphabetical order.

Slow Clock

This message appears when the active clock channel is not changing or is typically changing at 1 ms or slower intervals. Check for the following:

1. The R4000 system is powered on and running.
2. R4000 Support is selected in the 92A96 Configuration menu.
3. Custom clocking is selected in the Clock menu.
4. The clock and 8-channel probe connections between the interface housings and probe adapter are correct (clock, section names, and channel numbers match), are properly oriented (GND connects to ground), and are fully engaged.
5. The connections between the interface housings and 92A96 probe cables have matched color labels, matched slot numbers, and are properly keyed.
6. The connections between the 92A96 probe cables and probe connectors have matched color labels, matched slot numbers, and are properly keyed.
7. The orientation of position A1 on the R4000 microprocessor in the probe adapter is correct.
8. The orientation of position A1 on the probe adapter in the R4000 system is correct.
9. No bent or missing pins on the R4000 microprocessor or on either of the probe adapter sockets.
10. The Normal/Timing jumper is in the Normal position.

Waiting for Stop

This message appears when the trigger condition is satisfied and memory is full but the Manual Stop mode is selected in the Cluster Setup menu. The solution is to manually stop the DAS 9200 by pressing F1: STOP.

This message can also appear when other modules in the cluster have not filled their memories. Wait for the other modules to fill their memories. If the message does not disappear in a short time, press F1: STOP.

Waiting for Stop-Store

This message appears when the trigger condition is satisfied but the amount of post-fill memory specified in the trigger position field is not yet filled. Press F1: STOP to view the acquired data, then check for the following:

1. The trigger program in the Trigger menu is correct.
2. The storage qualification in the Trigger menu is correct.

Waiting for Trigger

This message appears when the trigger condition does not occur. Check for the following:

1. The R4000 system is powered on and running.
2. The trigger conditions are not being satisfied. The Module Monitor menu shows which state the trigger program is in. Press F1: STOP, access the Trigger menu, and redefine the conditions for that state. Also refer to the description on *Triggering* in Section 3.

OTHER PROBLEMS

There may be problems with disassembly for which no error messages are displayed. Some of these problems and their suggested solutions follow:

Incorrect Data

If the data acquired is obviously incorrect, check the following:

1. R4000 Support is selected in the 92A96 Configuration menu.
2. Custom clocking is selected in the 92A96 Clocking menu.
3. The Normal/Timing jumper is in the Normal position.
4. The clock and 8-channel probe connections between the interface housings and probe adapter are correct (clock, section names, and channel numbers match), are properly oriented (GND connects to ground), and are fully engaged.
5. The connections between the interface housings and 92A96 probe cables have matched color labels, matched slot numbers, and are properly keyed.
6. The connections between the 92A96 probe cables and probe connectors have matched color labels, matched slot numbers, and are properly keyed.
7. The orientation of position A1 on the R4000 microprocessor in the probe adapter is correct.
8. The orientation of position A1 on the probe adapter in the R4000 system is correct.
9. No bent or missing pins on the R4000 microprocessor or on either of the probe adapter sockets.

Other Suggestions

If the previous suggestions do not fix the problem with acquiring disassembled bus cycles from your R4000 system, try the following:

1. Reload the module setup by selecting another type of support (such as General Purpose) and then reselecting the R4000 Support in the 92A96 Configuration menu to restore the DAS 9200 to a known state.
2. Possible ac and dc loading problems may be remedied by removing one or both of the protective sockets from the probe adapter. These sockets may add enough inductance to your R4000 system to affect it. Refer to Appendix C for a description on how to remove sockets from the probe adapter.
3. If your R4000 system has a buffered TClock signal, with matching buffer between SyncOut and SyncIn, calibrate the TClock0/TClock1 jumpers on the probe adapter. For information on how to do this, refer to *TClock0 / TClock1 Jumpers* in Section 2.

If the DAS 9200 still is not acquiring data after trying these solutions, there may be a problem with your R4000 system. Try performing hardware analysis with your DAS 9200 system to ensure that the R4000 signals are valid at the time the probe adapter samples them.

Refer to *Section 4: Hardware Analysis,* for information on data sampling rates using either the Internal or External clocking selections in the Clock menu. Also refer to *Appendix B: How Data is Acquired* to see when the Custom clocking samples the various R4000 system signals.

Appendix B:
HOW DATA IS ACQUIRED

This appendix shows how the 92A96 Module acquires R4000 signals when used with Custom clocking. The 92A96 Module logs in all of the signals from a particular channel group at a point in the bus cycle when they are all valid.

CUSTOM CLOCKING

After the signals are logged in, they are sent to the trigger machine and the acquisition memory as a master sample (one complete data acquisition record). Figure B-1 shows the master sample point where the signals are logged in.

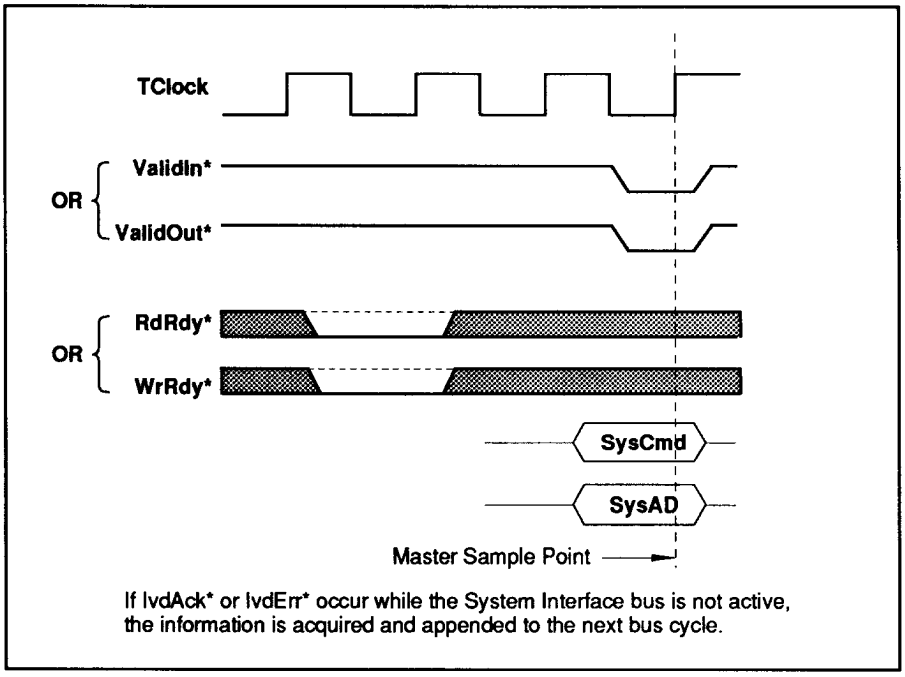


Figure B-1. R4000 Bus Timing. The 92DM75A samples signals when the 92DM75A-generated Master Clock occurs; the sampled signals are combined to form one complete acquisition record.

The probe adapter contains circuitry that derives the CmdRd*_D, CmdWr*_D, and Ivd*_D signals. It also contains circuitry that depipelines the RdRdy* and WrRdy* signals, and samples the ValidIn* and ValidOut* signals. In addition, it contains circuitry that logically OR's the six interrupt signals Int5*–Int0* and samples the resulting signal Int*_D.

CmdRd*_D is active (low) whenever a Read, Read-with-Write-Forthcoming, Invalidate, or Compulsory-Update command is encountered. CmdWr*_D is active (low) whenever a Write command is encountered. Ivd*_D is active (low) whenever either of the IvdAck* or IvdErr* signals are active (low).

When J1411 is placed in the Timing position, the signals RdRdy*, WrRdy*, ValidIn*, and ValidOut* are transparently passed from input to output with a delay of 5 ns. When J1411 is in the Timing position it also causes the CmdRd*_D, CmdWr*_D, Ivd*_D, and Int*_D signals to become inactive (float high).

SIGNALS NOT ACQUIRED

A single 92A96 Module does not have enough channels to acquire all the signals from the R4000 microprocessor. Table B-1 lists additional R4000 signals which are available at the auxiliary connector, J1281, on either the standard or option 2S probe adapters. You can disconnect the C0 8-channel probe and use the individual podlets to connect to these signals or other signals in the R4000 system.

Table B-1
Auxiliary R4000 Signals on J1281

Channel Name & Number	R4000 Signal Name	179-Pin PGA	447-Pin PGA	Channel Name & Number	R4000 Signal Name	179-Pin PGA	447-Pin PGA
AUX0:1	Ground	–	–	AUX0:9	JTCK	H17	U39
AUX0:2	MasterClock	J17	AA37	AUX0:10	JTMS	E16	G37
AUX0:3	MasterOut	P17	AJ39	AUX0:11	JTDI	G16	N39
AUX0:4	VccOk	M17	AE39	AUX0:12	Ground	–	–
AUX0:5	Fault*	B16	C39	AUX0:13	GrpRun*	U10	AV24
AUX0:6	Ground	–	–	AUX0:14	GrpStall*	T9	AV20
AUX0:7	Ground	–	–	AUX0:15	ModeIn	U4	AV8
AUX0:8	JTDO	F16	J39	AUX0:16	ModeClock	B4	B8

Table B-2 lists additional R4000SC and R4000MC signals which are only available on the other auxiliary connector, J1280, on the option 2S probe adapter.

Table B-2
Auxiliary R4000SC and R4000MC Signals on J1280

Channel Name & Number	R4000 Signal Name	447-Pin PGA	Channel Name & Number	R4000 Signal Name	447-Pin PGA
AUX1:1	Ground	–	AUX1:6	Status4	W37
AUX1:2	Status0	U33	AUX1:7	Status5	AC37
AUX1:3	Status1	U35	AUX1:8	Status6	AC35
AUX1:4	Status2	V36	AUX1:9	Status7	AC33
AUX1:5	Status3	W35	AUX1:10	Ground	–

Table B-3 lists the remaining R4000 signals which are not available for probing on either probe adapter. None of the 208 signals of the secondary cache of the R4000SC or R4000MC microprocessors are probed with the option 2S probe adapter.

Table B-3
R4000 Signals Not Available for Probing

R4000 Signal Name	179-Pin PGA	447-Pin PGA	R4000 Signal Name	179-Pin PGA	447-Pin PGA
SyncOut	P16	AN39	RClock0	T17	AM34
SyncIn	J16	W39	RClock1	R16	AL33
IOOut	U12	AV28	VccP	K17	AA33
IOIn	T13	AV32	VssP	K16	Y34

Warning

The following servicing instructions are for use only by qualified personnel. To avoid personnel injury, do not perform any servicing other than that contained in the operating instructions unless you are qualified to do so. Refer to General Safety Summary and Service Safety Summary prior to performing any service.

Appendix C:
SERVICE INFORMATION

This appendix contains the following information:

- safety summary
- brief description of the probe adapter and how it works
- care and maintenance
- specification tables
- channel assignment table
- dimensions of the probe adapter with all probes connected
- removing and replacing individual podlets
- removing and replacing sockets
- replaceable electrical parts list
- circuit board component location diagrams
- schematics
- replaceable mechanical parts list
- exploded mechanical diagram

SERVICING SAFETY INFORMATION

The following service safety information is for service technicians. Follow these safety precautions, along with the general precautions outlined in your DAS 9200 acquisition module user manual, while installing or servicing this product.

Do Not Service Alone. Do not perform internal service or adjustment on this product unless another person is present and able to give first aid and resuscitation.

Use Care When Servicing With Power On. To avoid personal injury from high current, remove jewelry such as rings, watches, and other metallic objects before servicing. Do not touch the product's exposed connections and components while power is on.

PROBE ADAPTER DESCRIPTION

The R4000 probe adapter is a non-intrusive piece of hardware that allows the DAS 9200 to acquire data from an R4000 microprocessor in its own operating environment with little effect, if any, on that system. The standard and option 2S probe adapters each consists of a circuit board with two PALs and a package of buffers. Each probe adapter connects to four clock probes and twelve 8-channel probes of a 92A96 acquisition module.

Signals from the R4000 system flow from the probe adapter to square pins connected to the 92A96 clock and 8-channel probes. Most signals then go across the 92A96 probe cables to the 92A96 module.

All circuitry on the probe adapter is powered from the SUT.

The standard R4000 probe adapter accommodates the R4000PC microprocessor in a 179-pin PGA package. The option 2S probe adapter accommodates the R4000SC or R4000MC microprocessors in a 447-pin PGA package. The probe adapter replaces the microprocessor in the SUT and the microprocessor is then placed back in the socket on the top of the probe adapter.

CARE AND MAINTENANCE

To maintain good electrical contact, keep the probe adapter free of dirt, dust and contaminants and ensure that any electrically conductive contaminants are removed.

Dirt and dust can usually be removed with a soft brush. For more extensive cleaning, use only a damp cloth. Abrasive cleaners and organic solvents should never be used.

CAUTION

The semiconductor devices contained on the probe adapter are susceptible to static-discharge damage. To prevent damage, service the probe adapter only in a static-free environment.

If the probe adapter is connected to your system, grasp the ground lug on the back of the DAS 9200 to discharge your stored static electricity. If the probe adapter is not connected, touch any of the ground pins (row of square pins closest to the edge of the probe adapter circuit board) to discharge stored static electricity from the probe adapter.

Always wear a grounding wrist strap, or similar device, while servicing the instrument.

CAUTION

Exercise care when soldering on a multilayer circuit board. Excessive heat can damage the through-hole plating or lift a run or pad and damage the board beyond repair. Do not apply heat for longer than three seconds. Do not apply heat consecutively to adjacent leads. Allow a moment for the board to cool between each operation.

If you must replace an electrical component on a circuit board, exercise extreme caution while desoldering or soldering the new component. Use a pencil-type soldering iron of less than 18 watts and an approved desoldering tool. Ensure that the replacement is an equivalent part by comparing the description as listed in the replaceable parts list.

SPECIFICATIONS

These specifications are for an R4000 probe adapter connected to a 92A96 Data Acquisition Module and the SUT. Table C-1 shows the electrical requirements the system under test must produce for the probe adapter to acquire correct data. Table C-2 shows the physical specifications for the probe adapter. Table C-3 shows the environmental specifications.

Table C-1
Electrical Specifications

Characteristics	Loading	
Maximum SUT Signal Loading	**AC Load**	**DC Load**
SysAD63–SysAD0	6 pF + 1 podlet	1 podlet
SysCmd8–SysCmd0	15 pF + 1 podlet	PAL + 1 podlet
Control		
IvdAck*, IvdErr*	15 pF + 1 podlet	PAL + 1 podlet + 10K to Vcc
RdRdy*, WrRdy*, ValidIn*, ValidOut*, Int5*–Int0*	15 pF	PAL
NMI*, Reset*, ColdReset*, Release*, SysADC7–SysADC0, SysCmdP, ExtRqst*	6 pF + 1 podlet	1 podlet
TClock1	60 pF†	74FR244 + 2 PALs‡
TClock0	60 pF + 2 podlets†	74FR244 + 2 podlets‡
All other	3 pF	none

	Requirements	
SUT DC Power Required		
Voltage	+5 V ± 0.25 V	
Current	< 500 mA calculated max	
SUT Clock		
Clock Rate	50 MHz max.	
	92DM75A Probe Adapter	**System Interface Bus**
TClock Frequency	50 MHz max. §	50 MHz max. ¶
TClock Period (tTCP)	20 ns min.§	20 ns to 40 ns¶
TClock Jitter (tTCJ)	none specified	± 1.0 ns¶
Minimum Setup Time Required#		
Data (tDS–tTCJ)	5.0 ns	5.0 ns ± 1.0 ns¶
Minimum Hold Time Required#		
Data (tDH–tTCJ)	0 ns	2.0 ns ± 1.0 ns¶

† When J1412 and J1421 are not in the N position (pins 1 and 2 connected), AC loading will drop to 35 pF.
‡ When J1412 and J1421 are in the N position (pins 1 and 2 connected), these lines are each terminated with 2 100 Ω 15 pF components in series to ground. In any other position, these lines are terminated with 1 such RC network.
§ Specification at time of printing. Contact your DAS 9200 sales representative for information on the fastest devices supported.
¶ These are worse case values.
# If the SUT has an external buffer between SyncOut and SyncIn, the TClock occurs further ahead of data transitions. This causes a decrease in setup time and an increase in hold time. If this happens, you need to calibrate the probe adapter to adjust its TClock with the buffered TClock in the SUT. Refer to the description on configuring the probe adapter in Section 2 of this manual for information on how to make this adjustment.

Table C-2
Physical Specifications

Characteristics	Description	
Overall Dimensions with Podlets Attatched (maximum)		
Probe Adapter	in	cm
Standard (R4000PC)		
Length	3.75	≈ 9.5
Width	2.90	≈ 7.4
Height†	2.24	≈ 5.7
Option 2S (R4000SC/R4000MC)		
Length	4.125	≈ 10.5
Width	3.15	≈ 8.0
Height†	2.24	≈ 5.7
Cable Length		
92A96 probe cable from the DAS 9200 to the tip of the 8-channel probes	5 ft 10.5 in	≈ 179 cm
Weight, probe adapter		
Standard	2.5 oz	(≈ 70 g)
Option 2S	3.0 oz	(≈ 84 g)

† With one socket extender excluding length of pins into SUT socket.

Table C-3
Environmental Specifications

Characteristic	Description
Temperature	
Max. Operating	+50° C (+122° F)†
Min. Operating	+0° C (+32° F)
Non-Operating	–62° C to +85° C (–78° F to +185° F)
Humidity	0 - 95 % relative humidity ‡
Altitude	
Operating	15,000 ft. (4.5 km) maximum
Non-Operating	50,000 ft. (15 km) maximum
Electrostatic Immunity	The probe adapter is static-sensitive

† Not to exceed R4000 thermal considerations. Cooling, using forced air, may be required across the CPU.
‡ Tested to Tektronix Standard 062-2847-00 Class 5.

Table C-4 shows the 92A96 section and channel assignments, their grouping and radixes, the voltage threshold, polarity, and microprocessor signal and pin connections.

Table C-4
92DM75A Channel Assignments

92DM75A Group (Radix)	Group Bit Pos	92A96 Section: Channel	Voltage Threshold, Polarity	R4000 Signal Name	179-Pin PGA	447-Pin PGA
SysAD-Hi	31	A3: 7	TTL, +	SysAD63	N17	AG39
(Hex)	30	A3: 6	TTL, +	SysAD62	N16	AL39
	29	A3: 5	TTL, +	SysAD61	U17	AR39
	28	A3: 4	TTL, +	SysAD60	T15	AU37
	27	A3: 3	TTL, +	SysAD59	V15	AW35
	26	A3: 2	TTL, +	SysAD58	U13	AW31
	25	A3: 1	TTL, +	SysAD57	T11	AW27
	24	A3: 0	TTL, +	SysAD56	T10	AW23
	23	A2: 7	TTL, +	SysAD55	T7	AW15
	22	A2: 6	TTL, +	SysAD54	T6	AW11
	21	A2: 5	TTL, +	SysAD53	U3	AW5
	20	A2: 4	TTL, +	SysAD52	T3	AU3
	19	A2: 3	TTL, +	SysAD51	R2	AR1
	18	A2: 2	TTL, +	SysAD50	N3	AM2
	17	A2: 1	TTL, +	SysAD49	M3	AJ1
	16	A2: 0	TTL, +	SysAD48	L2	AE1
	15	A1: 7	TTL, +	SysAD47	F17	M38
	14	A1: 6	TTL, +	SysAD46	E17	H38
	13	A1: 5	TTL, +	SysAD45	B17	E39
	12	A1: 4	TTL, +	SysAD44	C15	C37
	11	A1: 3	TTL, +	SysAD43	A15	A35
	10	A1: 2	TTL, +	SysAD42	B13	A31
	9	A1: 1	TTL, +	SysAD41	C11	A27
	8	A1: 0	TTL, +	SysAD40	C10	A23
	7	A0: 7	TTL, +	SysAD39	C7	A15
	6	A0: 6	TTL, +	SysAD38	C6	A11
	5	A0: 5	TTL, +	SysAD37	B3	A5
	4	A0: 4	TTL, +	SysAD36	C3	C3
	3	A0: 3	TTL, +	SysAD35	D2	E1
	2	A0: 2	TTL, +	SysAD34	F3	H2
	1	A0: 1	TTL, +	SysAD33	G3	L1
	0	A0: 0	TTL, +	SysAD32	H2	R1

(Cont.)

Table C-4 (Cont.)
92DM75A Channel Assignments

92DM75A Group (Radix)	Group Bit Pos	92A96 Section: Channel	Voltage Threshold, Polarity	R4000 Signal Name	179-Pin PGA	447-Pin PGA
SysAD-Lo	31	D3: 7	TTL, +	SysAD31	M16	AH38
(Hex)	30	D3: 6	TTL, +	SysAD30	R17	AM38
	29	D3: 5	TTL, +	SysAD29	T16	AR37
	28	D3: 4	TTL, +	SysAD28	U15	AV38
	27	D3: 3	TTL, +	SysAD27	U14	AW33
	26	D3: 2	TTL, +	SysAD26	T12	AW29
	25	D3: 1	TTL, +	SysAD25	U11	AW25
	24	D3: 0	TTL, +	SysAD24	U9	AW21
	23	D2: 7	TTL, +	SysAD23	U6	AW13
	22	D2: 6	TTL, +	SysAD22	U5	AW9
	21	D2: 5	TTL, +	SysAD21	T4	AW3
	20	D2: 4	TTL, +	SysAD20	T2	AU1
	19	D2: 3	TTL, +	SysAD19	P3	AN3
	18	D2 2	TTL, +	SysAD18	P1	AL3
	17	D2: 1	TTL, +	SysAD17	M2	AH2
	16	D2: 0	TTL, +	SysAD16	K2	AD2
	15	D1: 7	TTL, +	SysAD15	E18	L39
	14	D1: 6	TTL, +	SysAD14	D17	G39
	13	D1: 5	TTL, +	SysAD13	C16	E37
	12	D1: 4	TTL, +	SysAD12	B15	B38
	11	D1: 3	TTL, +	SysAD11	B14	A33
	10	D1: 2	TTL, +	SysAD10	C12	A29
	9	D1: 1	TTL, +	SysAD9	B11	A25
	8	D1: 0	TTL, +	SysAD8	B9	A21
	7	D0: 7	TTL, +	SysAD7	B6	A13
	6	D0: 6	TTL, +	SysAD6	B5	A9
	5	D0: 5	TTL, +	SysAD5	C4	A3
	4	D0: 4	TTL, +	SysAD4	C2	C1
	3	D0: 3	TTL, +	SysAD3	E3	G3
	2	D0: 2	TTL, +	SysAD2	E1	J3
	1	D0: 1	TTL, +	SysAD1	G2	M2
	0	D0: 0	TTL, +	SysAD0	J2	T2

(Cont.)

Table C-4 (Cont.)
92DM75A Channel Assignments

92DM75A Group (Radix)	Group Bit Pos	92A96 Section: Channel	Voltage Threshold, Polarity	R4000 Signal Name	179-Pin PGA	447-Pin PGA
Control	11	C3: 3	TTL, +	Reset*	U16	AU39
(Sym)	10	C2: 3	TTL, +	ValidIn*	P2	AN1
(R4000_Ctrl)	9	C2: 7	TTL, +	ValidOut*	R3	AR3
	8	C3: 6	TTL, +	SysCmd8	C13	B32
	7	C3: 5	TTL, +	SysCmd7	B12	B28
	6	C3: 4	TTL, +	SysCmd6	B10	B24
	5	C3: 2	TTL, +	SysCmd5	C9	B20
	4	C3: 1	TTL, +	SysCmd4	B7	B16
	3	C3: 0	TTL, +	SysCmd3	A5	B12
	2	C2: 6	TTL, +	SysCmd2	B2	B2
	1	C2: 5	TTL, +	SysCmd1	D3	E3
	0	C2: 4	TTL, +	SysCmd0	E2	G1
Intr (Sym) (R4000_Intr)	1	C1: 7	TTL, +	Int5*–Int0* OR'ed to make Int*_D†	F2, H3, J3, K3, L3, N2	AL1 (Int*0)
	0	C1: 6	TTL, +	NMI*	U7	AV16
Ack	1	C1: 4	TTL, +	IvdErr*‡	none	AA39
(Sym) (R4000_Ack)	0	C1: 0	TTL, +	IvdAck*‡	none	AA35
Misc	3	C3: 7	TTL, +	TClock0§	C17	H34
(Off)	2	C2: 2	TTL, +	ValidOut*=§	R3	AR3
	1	C2: 1	TTL, +	RdRdy*_L§	T5	AW7
	0	C2: 0	TTL, +	WrRdy*_L§	C5	A7
Misc1	2	C1: 3	TTL, +	ColdReset*	T14	AW37
	1	C1: 5	TTL, +	ExtRqst*§	U2	AV2
(Off)	0	C1: 1	TTL, +	Release*§	V5	AV12
Misc2	8	C1: 2	TTL, +	SysCmdP	C14	A37
(off)	7	C0: 7	TTL, +	SysADC7§	L17	AC39
	6	C0: 6	TTL, +	SysADC6§	U8	AW19
	5	C0: 5	TTL, +	SysADC5§	H16	T38
	4	C0: 4	TTL, +	SysADC4§	B8	A19
	3	C0: 3	TTL, +	SysADC3§	L16	AD38
	2	C0: 2	TTL, +	SysADC2§	T8	AW17
	1	C0: 1	TTL, +	SysADC1§	G17	R39
	0	C0: 0	TTL, +	SysADC0§	C8	A17

(Cont.)

Table C-4 (Cont.)
92DM75A Channel Assignments

92DM75A Group (Radix)	Group Bit Pos	92A96 Section: Channel	Voltage Threshold, Polarity	R4000 Signal Name	179-Pin PGA	447-Pin PGA
Qualifiers	3	C2: 3	TTL, +	ValidIn*	P2	AN1
	2	C2: 2	TTL, +	ValidOut*=§	R3	AR3
	1	C2: 1	TTL, +	RdRdy*_L§	T5	AW7
	0	C2: 0	TTL, +	WrRdy*_L§	C5	A7
Clock	3	CLK: 3	TTL, +	TClock0=	C17	H34
(not	2	CLK: 2	TTL, +	Ivd*_D†	none	none
stored)	1	CLK: 1	TTL, +	CmdRd*_D†	none	none
	0	CLK: 0	TTL, +	CmdWr*_D†	none	none

† These signals are derived.
‡ The Invalidate/Update Acknowledge signals on the R4000MC occur asynchronously to the system interface bus. The 92DM75A acquires and appends this information to the following bus cycle. The default symbol table indicates which one was received.
§ Data acquired on these signals is only valid when using Internal or External clocking.

Figure C-1 shows the dimensions of the standard probe adapter with all the clock probes and 8-channel probes attached.

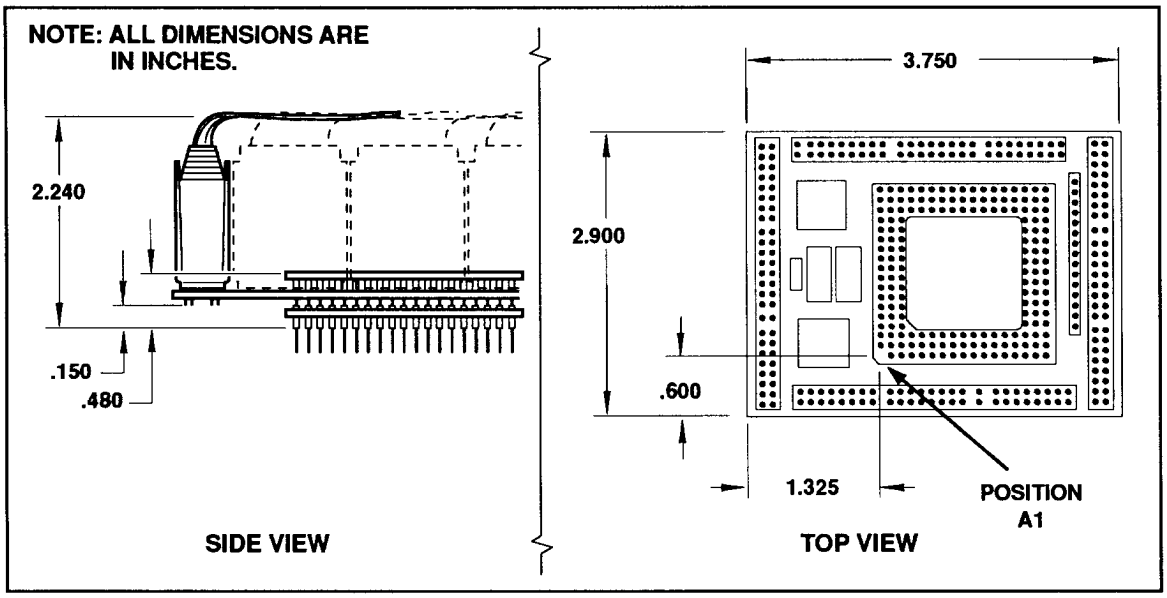


Figure C-1. Minimum clearance of the standard probe adapter with probes attached.

Figure C-2 shows the dimensions of the option 2S probe adapter with all the clock probes and 8-channel probes attached.

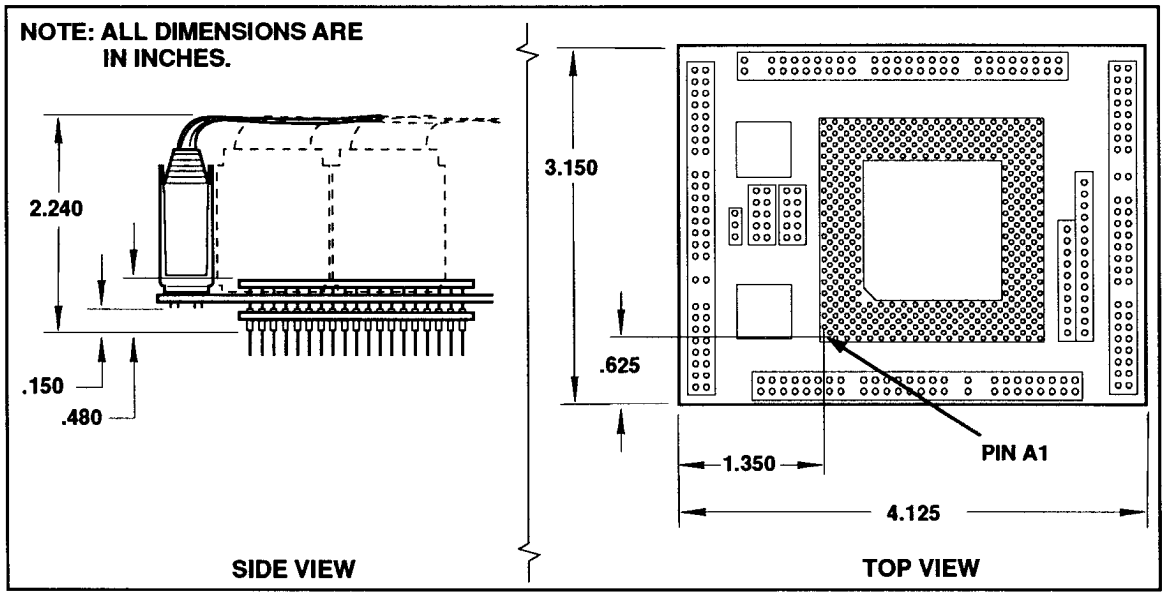


Figure C-2. Minimum clearance of the option 2S probe adapter with probes attached.

DISCONNECTING CLOCK AND 8-CHANNEL PROBES

You may need to disconnect the clock and 8-channel probes from the probe adapter to use them on another application, to connect individual podlets to other signals in your R4000 system, or to replace defective clock or probe channels (podlets). Refer to Figure C-3 and the following procedure to disconnect the clock and 8-channel probes from the probe adapter. Use the antistatic shipping material to support the probe adapter while disconnecting the clock and 8-channel probes.

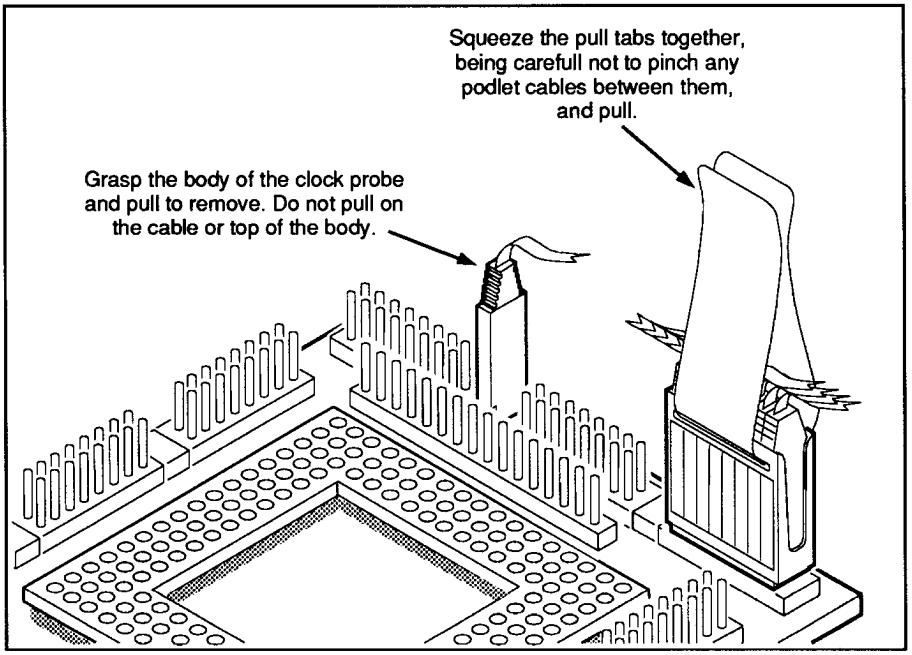


Figure C-3. Disconnecting clock and 8-channel probes.

1. Power down the SUT. It is not necessary to power down the DAS 9200.

CAUTION

Pulling on the cables, on the neck of the clock probe, or pinching the cables between the pull tabs can damage the probes. Always handle the probes by their bodies.

2. Firmly grasp the body of a clock probe and gently pull it off of the square pins.
3. Squeeze the pull tabs on the podlet holder together, be careful not to pinch any podlet cables between them.
4. Gently pull the 8-channel probe off of the square pins.

REMOVING AND REPLACING PODLETS

Each 8-channel probe consists of 8 single-channel podlets ganged together in a podlet holder. You may need to remove these podlets from an 8-channel probe to use for alternate signal connections on the probe adapter or in the R4000 system.

Most signals are used for Disassembly display (Custom clocking). Refer to Appendix B for information on the probes that can be connected to other signals on the probe adapter or in the R4000 system. If viewing the Timing or State menus for hardware analysis (internal or external clocking), any signal may be used as desired.

You can also use these procedures to replace a defective clock probe or a defective podlet from an 8-channel probe.

Removing a Clock Probe or 8-Channel Probe Podlet from the Interface Housing

Refer to Figure C-4 and the following procedure to remove a clock probe or an 8-channel probe podlet from the interface housing.

1. Power down the SUT. It is not necessary to power down the DAS 9200.
2. Use a small pointed tool such as a ballpoint pen, pencil, or straightened paper clip to press down on the latch detent of the podlet through an opening on the interface housing.
3. Gently pull the podlet connector out of the housing with one hand while pressing down on the latch detent with the pointed tool.

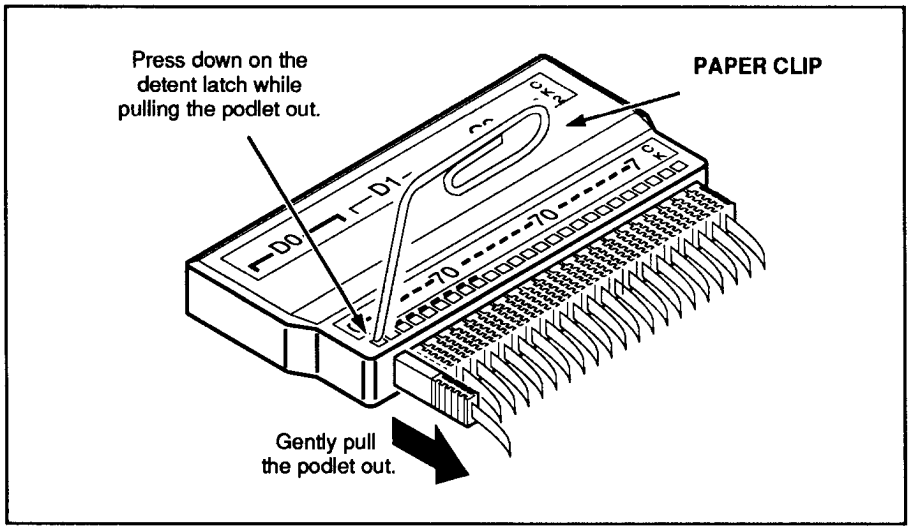


Figure C-4. Removing a clock or an 8-channel probe podlet from the interface housing.

Replacing a Clock Probe

To replace a clock probe, insert a new clock probe into the same clock channel position on the interface housing. Insert the clock probe into the interface housing with the detent latch oriented to the label side of the housing. Refer to Figure C-4.

Removing 8-Channel Probe Podlets from the Podlet Holder

Refer to Figure C-5 and the following procedure to remove the 8-channel probe podlets from the podlet holder.

CAUTION

Excessive pulling on the sides of the holder can damage the podlet holder; spread the holder open wide enough to clear and remove the podlets.

1. To remove podlets from the podlet holder, grasp the plastic pull tab on each side of the podlet holder and gently spread the sides of the holder open just enough to clear a podlet.
2. Remove the middle two podlets from the podlet holder by pushing up on the metal pin receptacles.
3. Release the tabs on the podlet holder.
4. Remove the remaining podlets by turning and extracting each one at a time.

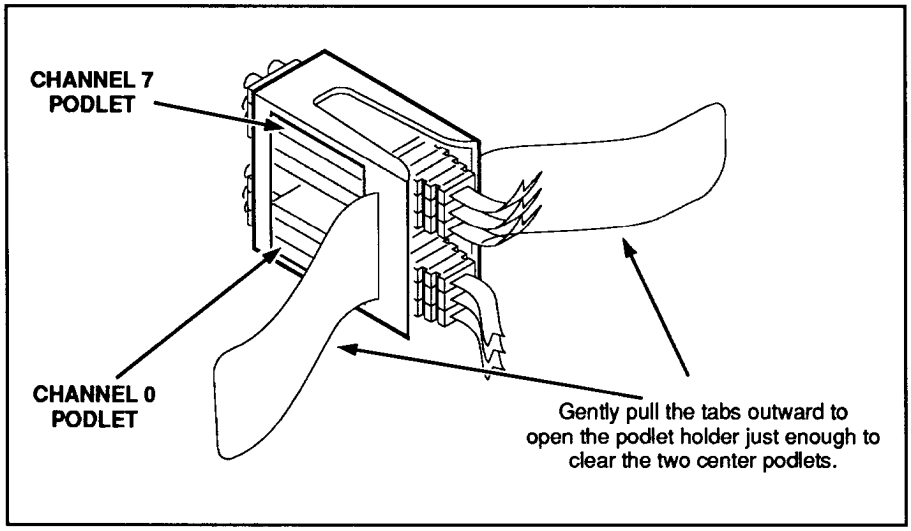


Figure C-5. Ganging together the 8-channel probe podlets.

Replacing 8-Channel Probe Podlets

The channel podlets must retain the same channel order on both the interface housing and in the podlet holder. Be sure to replace the old podlet with a podlet of the same color. Table C-5 shows the color code and channel number of each podlet for an 8-channel probe.

Table C-5
Podlet-to-Channel Color Code

Podlet Color	Channel
Black	0
Brown	1
Red	2
Orange	3
Yellow	4
Green	5
Blue	6
Violet	7

Refer to Figure C-5 and the following procedure to replace an 8-channel probe podlet.

1. Insert the appropriately-colored podlet into the interface housing with the detent latch oriented to the label side of the housing.
2. If you are replacing a single podlet, orient the podlet connector marked GND towards the side of the podlet holder labeled GROUND.
3. Grasp the plastic pull tab on each side of the holder and gently spread the sides of the holder open just enough to clear the podlet.
4. Hold the podlet body with the other hand and place it in the holder in the correct channel order. Do not grasp and turn the podlet cable.
5. If you are re-ganging all the podlets of an 8-channel probe, begin ganging the podlets together starting with either channel 0 or channel 7. Orient the podlet channel marked GND towards the side of the podlet holder labeled GROUND.

CAUTION

To prevent damage to the podlets, keep the podlet cables parallel to each other when ganging them into the holder. Avoid twisting the podlet cables between the interface housing and the podlet holder.

6. Hold the podlet body, turn the podlet body parallel to the sides of the holder, move it into the holder, and use your fingers to press it into place perpendicular to the sides of the holder. Be sure to gang the podlets in the correct channel order according to the channel label on the podlet holder and podlet color code, with all ground channels toward the Ground side of the holder. Do not place the podlet into the holder by grasping the podlet cable.
7. Continue placing the next two podlets, one at a time, in channel order, in the podlet holder. Orient all ground channels toward the Ground side of the holder.
8. The fourth podlet should be either channel 0 or 7, whichever one is not already placed in the holder. Place this podlet in the other end of the podlet holder and orient the ground channel correctly.
9. Continue placing the next two podlets, one at a time, in channel order, in the podlet holder. Continue orienting the ground channels correctly.

CAUTION

Excessive pulling on the sides can break the podlet holder. Spread the holder open only wide enough to clear the podlet.

10. Grasp the plastic pull tab on each side of the holder and gently spread the sides of the holder open just enough to clear a podlet.
11. Place the last pair of podlets (channels 3 and 4) in the podlet holder in proper channel order, orienting the ground channels to the Ground side of the holder.

REMOVING AND REPLACING SOCKETS

The probe adapter board contains several sockets designed both to protect the probe adapter and to make it easy to insert and remove the R4000 microprocessor. Figure C-6 shows a side view of the board, sockets, and pins of the probe adapter. The fixed socket on top of the probe adapter board is soldered and cannot be removed. The following paragraphs describe how to remove and replace the protective socket on the bottom of the probe adapter board.

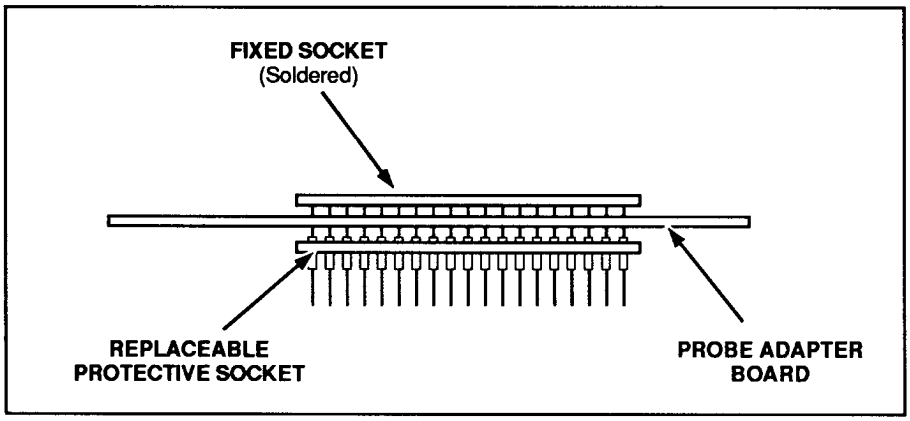


Figure C-6. Side view of the R4000 Probe Adapter board.

Replaceable Protective Socket

You should not have to remove the probe adapter's replaceable protective socket unless there is socket-pin damage. To remove the protective socket, refer to Figure C-6 and follow these steps:

1. Locate a corner of the socket adjacent to the edge of the probe adapter board.

NOTE

Be careful not to damage any etched circuit board runs or components surrounding the protective socket when using a pair of tweezers to remove the protective socket from the bottom of the probe adapter board.

2. Position a pair of curve-tipped tweezers (such as Miltex PL317) at a 45° angle to the corner of the protective socket. The curved side of the tweezers should be toward the board. Figure C-7 shows how to position the tweezers.
3. With the tips of the tweezers open, align them with the spaces between the socket pins. Interpin spaces are diagonal to the outside edge of the socket. Slide the tips of the tweezers in between the pins until the curved edge is just under the socket.

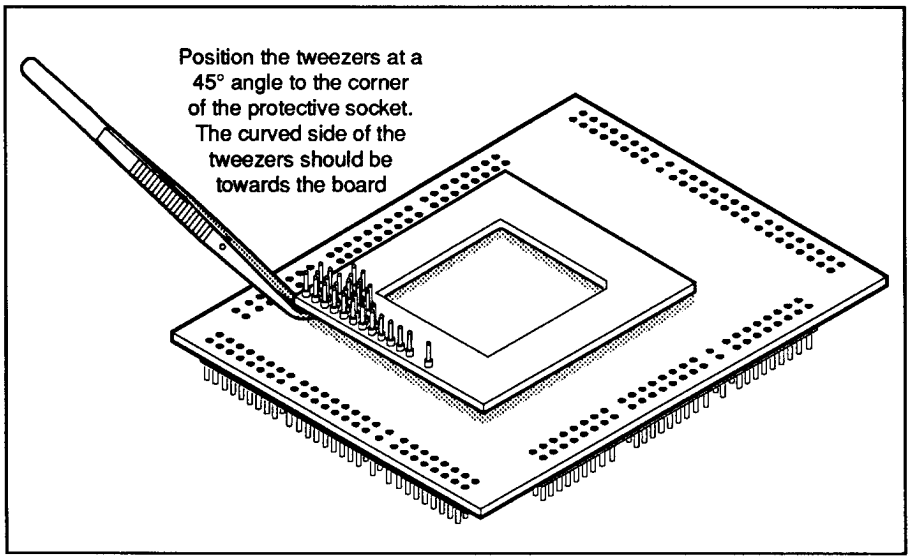


Figure C-7. Tool position for socket removal.

4. Without closing the tweezers, gently press down on the handle leveraging the socket up until the socket begins to separate from the probe adapter board pins.

CAUTION

Do not completely pry off one side of the protective socket and then the other. Applying uneven pressure can damage the socket's pins. Do not use board components as leverage to remove the socket.

5. Use the tweezers on the other three corners of the socket, but only where there are no board runs, prying the socket loose a little at a time. Use even pressure alternately on all corners of the socket until the socket is completely loose.
6. Remove the socket from the board.

To replace the protective socket, follow these steps:

1. Check that the new socket's pins are straight.
2. Place the socket on the pins of the probe adapter board; make sure that all pins line up correctly.
3. Press the socket onto the board by pressing them against a hard, flat surface while applying even pressure.

Replaceable Electrical Parts

Parts Ordering Information

Replacement parts are available from or through your local Tektronix, Inc. Field Office or representative.

When ordering parts, include the following information in your order: part number, instrument type or number, serial number, and modification number if applicable.

If a part you have ordered has been replaced with a new or improved part, your local Tektronix, Inc. Field Office or representative will contact you concerning any change in part number.

Change information, if any, is located at the rear of this manual.

List of Assemblies

A list of assemblies can be found at the beginning of the electrical parts list. The assemblies are listed in numerical order. When the complete component number of a part is known, this list will identify the assembly in which the part is located.

Cross Index-Mfr. Code Number to Manufacturer

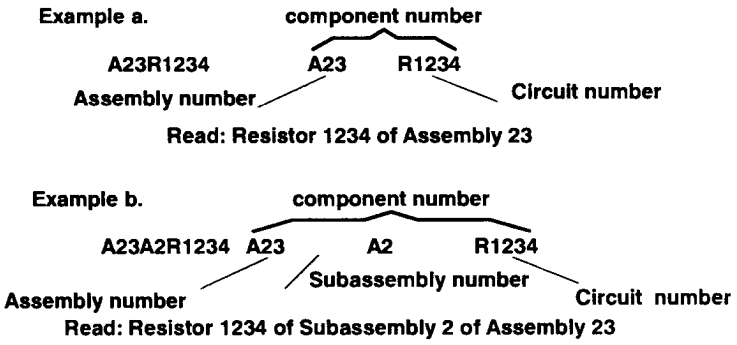
The Mfg. Code Number to Manufacturer Cross Index for the electrical parts list is located immediately after this page. The cross index provides codes, names, and addresses of manufacturers of components listed in the electrical parts list.

Abbreviations

Abbreviations conform to American National Standard Y1.1.

Component Number

(column 1 of the parts list)



The circuit component's number appears on the diagrams and circuit board illustrations. Each diagram and circuit board illustration is clearly marked with the assembly number. Assembly numbers are also marked on the mechanical exploded views located in the mechanical parts list. The component number is obtained by adding the assembly number prefix to the circuit number.

The electrical parts list is divided and arranged by assemblies in numerical sequence (e.g., assembly A1 with its subassemblies and parts, precedes assembly A2 with its subassemblies and parts).

Chassis-mounted parts have no assembly number prefix and are located at the end of the electrical parts list.

Tektronix Part No.

(column 2 of the parts list)

Indicates part number to be used when ordering replacement part from Tektronix.

Serial No.

(columns 3 & 4 of the parts list)

Column three (3) indicates the serial number at which the part was first used. Column four (4) indicates the serial number at which the part was removed. No serial number entered indicates part is good for all serial numbers.

Name & Description

(column five of the parts list)

In the parts list, an item name is separated from the description by a colon (:). Because of space limitations, an item name may sometimes appear as incomplete. For further item name identification, the U.S. Federal Catalog handbook H6-1 can be utilized where possible.

Mfr. Code

(column 6 of the parts list)

Indicates the code number of the actual manufacturer of the part. (Code to name and address cross reference can be found immediately after this page.)

Mfr. Part No.

(column 7 of the parts list)

Indicates actual manufacturer's part number.

CROSS INDEX – MFR. CODE NUMBER TO MANUFACTURER

Mfr. Code	Manufacturer	Address	City, State, Zip Code
00779	AMP INC	2800 FULLING MILL PO BOX 3608	HARRISBURG PA 17105
04222	AVX CERAMICS DIV OF AVX CORP	19TH AVE SOUTH P O BOX 867	MYRTLE BEACH SC 29577
26742	METHODE ELECTRONICS INC	7447 W WILSON AVE	CHICAGO IL 60656–4548
27014	NATIONAL SEMICONDUCTOR CORP	2900 SEMICONDUCTOR DR	SANTA CLARA CA 95051–0606
50139	ALLEN–BRADLEY CO ELECTRONIC COMPONENTS	1414 ALLEN BRADLEY DR	EL PASO TX 79936
58050	TEKA PRODUCTS INC	45 SALEM ST	PROVIDENCE RI 02907
63058	MCKENZIE TECHNOLOGY	44370 OLD WARMS SPRINGS BLVD	FREMONT CA 94538
80009	TEKTRONIX INC	14150 SW KARL BRAUN DR PO BOX 500	BEAVERTON OR 97077–0001

Component Number	Tektronix Part No.	Serial No. Effective	Dscont	Name & Description	Mfr. Code	Mfr. Part No.
A01	671-2652-00			CIRCUIT BD ASSY:R4000PC,PROBE ADAPTER	80009	671265200
A02	671-2772-00			CIRCUIT BD ASSY:R4000SC/MC,PROBE ADAPTER	80009	671277200
A01	671-2652-00			CIRCUIT BD ASSY:R4000PC,PROBE ADAPTER	80009	671265200
A01C1111	283-5187-00			CAP,FXD,CERAMIC:MLC;15PF,5%,100V,NPO,1206	04222	12061A0150JAT1A
A01C1131	283-5187-00			CAP,FXD,CERAMIC:MLC;15PF,5%,100V,NPO,1206	04222	12061A0150JAT1A
A01C1211	283-5004-00			CAP,FXD,CERAMIC:MLC;15PF,5%,100V,NPO,1206	04222	12061A0150JAT1A
A01C1241	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A01C1255	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A01C1261	283-5004-00			CAP,FXD,CERAMIC:MLC;0.1UF,10%,25V,X7R,1206	04222	12063C104KAT3A
A01C1271	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A01C1450	283-5187-00			CAP,FXD,CERAMIC:MLC;15PF,5%,100V,NPO,1206	04222	12061A0150JAT1A
A01C1451	283-5004-00			CAP,FXD,CERAMIC:MLC;0.1UF,10%,25V,X7R,1206	04222	12063C104KAT3A
A01C1461	283-5187-00			CAP,FXD,CERAMIC:MLC;15PF,5%,100V,NPO,1206	04222	12061A0150JAT1A
A01C1611	283-5004-00			CAP,FXD,CERAMIC:MLC;0.1UF,10%,25V,X7R,1206	04222	12063C104KAT3A
A01C1651	283-5004-00			CAP,FXD,CERAMIC:MLC;0.1UF,10%,25V,X7R,1206	04222	12063C104KAT3A
A01C1711	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A01C1841	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A01C1851	283-5004-00			CAP,FXD,CERAMIC:MLC;0.1UF,10%,25V,X7R,1206	04222	12063C104KAT3A
A01C1881	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A01C1882	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A01J1102	---------			CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (SEE RMPL FIG C-8)	80009	131526700
A01J1181	---------			CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (SEE RMPL FIG C-8)	80009	131526700
A01J1281	---------			CONN,HDR:PCB,;MALE,STR,1 X 36,0.1 CTR,0.230 (SEE RMPL FIG C-8)	58050	082-3644-SS10
A01J1281	---------			CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (SEE RMPL FIG C-8)	80009	131526700
A01J1411	---------			CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (SEE RMPL FIG C-8)	80009	131526700
A01J1412	---------			CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (SEE RMPL FIG C-8)	80009	131526700
A01J1421	---------			CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (SEE RMPL FIG C-8)	80009	131526700
A01J1911	---------			CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (SEE RMPL FIG C-8)	80009	131526700
A01J1991	---------			CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (SEE RMPL FIG C-8)	80009	131526700
A01P1411	---------			CONN,BOX:SHUNT/SHORTING,;FEMALE,1 X 2,0.1(SEE RMPL FIG C-8)	26742	9618-302-50
A01P1412	---------			CONN,BOX:SHUNT/SHORTING,;FEMALE,1 X 2,0.1(SEE RMPL FIG C-8)	26742	9618-302-50

Component Number	Tektronix Part No.	Serial No. Effective	Dscont	Name & Description	Mfr. Code	Mfr. Part No.
A01P1421	----------			CONN,BOX:SHUNT/SHORTING,;FEMALE,1 X 2,0.1(SEE RMPL FIG C-8)	26742	9618-302-50
A01R1111	321-5006-00			RES,FXD:THICK FILM;100 OHM,1%,0.125W,TC=100	50139	BCK1000FT
A01R1131	321-5006-00			RES,FXD:THICK FILM;100 OHM,1%,0.125W,TC=100	50139	BCK1000FT
A01R1450	321-5006-00			RES,FXD:THICK FILM;100 OHM,1%,0.125W,TC=100	50139	BCK1000FT
A01R1461	321-5006-00			RES,FXD:THICK FILM;100 OHM,1%,0.125W,TC=100	50139	BCK1000FT
A01R1811	321-5006-00			RES,FXD:THICK FILM;100 OHM,1%,0.125W,TC=100	50139	BCK1000FT
A01R1811	321-5030-00			RES,FXD:THICK FILM;10.0K OHM,1%,0.125W,TC=100 PPM	50139	BCK1002FT
A01R1821	321-5030-00			RES,FXD:THICK FILM;10.0K OHM,1%,0.125W,TC=100 PPM	50139	BCK1002FT
A01R1831	321-5030-00			RES,FXD:THICK FILM;10.0K OHM,1%,0.125W,TC=100 PPM	50139	BCK1002FT
A01U1321	160-9148-00			IC,DIGITAL:STTL,PLD;PAL20R6,156-6381-00	80009	160-9148-00
A01U1721	160-9149-00			IC,DIGITAL:STTL,PLD;PAL20R6,156-6381-00	80009	160-9149-00
A01U1831	----------			SOCKET,PGA:PCB,R4000;179 POS,18 X 18,0.1 CT (SEE RMPL C-8)	63058	PGA0179H101B1-18
A01U2541	156-6401-00			IC,DIGITAL:FTTL,BUFFER;NON INV OCTAL LINE	27014	74FR244SC

Component Number	Tektronix Part No.	Serial No. Effective	Serial No. Dscont	Name & Description	Mfr. Code	Mfr. Part No.
A02	671-2772-00			CIRCUIT BD ASSY:R4000SC/MC,PROBE ADAPTER	80009	671277200
A02C1111	283-5187-00			CAP,FXD,CERAMIC:MLC;15PF,5%,100V,NPO,1206	04222	12061A0150JAT1A
A02C1121	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A02C1131	283-5187-00			CAP,FXD,CERAMIC:MLC;15PF,5%,100V,NPO,1206	04222	12061A0150JAT1A
A02C1141	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A02C1161	283-5004-00			CAP,FXD,CERAMIC:MLC;0.1UF,10%,25V,X7R,1206	04222	12063C104KAT3A
A02C1181	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A02C1255	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A02C1311	283-5004-00			CAP,FXD,CERAMIC:MLC;0.1UF,10%,25V,X7R,1206	04222	12063C104KAT3A
A02C1421	283-5004-00			CAP,FXD,CERAMIC:MLC;0.1UF,10%,25V,X7R,1206	04222	12063C104KAT3A
A02C1450	283-5187-00			CAP,FXD,CERAMIC:MLC;15PF,5%,100V,NPO,1206	04222	12061A0150JAT1A
A02C1451	283-5004-00			CAP,FXD,CERAMIC:MLC;0.1UF,10%,25V,X7R,1206	04222	12063C104KAT3A
A02C1461	283-5187-00			CAP,FXD,CERAMIC:MLC;15PF,5%,100V,NPO,1206	04222	12061A0150JAT1A
A02C1711	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A02C1841	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A02C1850	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A02C1851	283-5004-00			CAP,FXD,CERAMIC:MLC;0.1UF,10%,25V,X7R,1206	04222	12063C104KAT3A
A02C1881	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A02C1882	283-5003-00			CAP,FXD,CERAMIC:MLC;0.01UF,10%,50V,X7R,1206	04222	12065C103KAT1A
A02J1102	131-5267-00			CONN,HDR:PCB,;MALE,STR,1 X 36,0.1 CTR,0.230 (SEE RMPL C-9)	80009	131526711
A02J1181	----------			CONN,HDR:PCB,;MALE,STR,1 X 36,0.1 CTR,0.230 (SEE RMPL C-9)		
A02J1280	----------			CONN,HDR:PCB,;MALE,STR,1 X 36,0.1 CTR,0.230 (SEE RMPL C-9)		
A02J1281	----------			CONN,HDR:PCB,;MALE,STR,1 X 36,0.1 CTR,0.230 (SEE RMPL C-9)		
A02J1411	----------			CONN,HDR:PCB,;MALE,STR,1 X 3,0.1 CTR,0.230(SEE RMPL C-9)		
A02J1412	----------			CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235(SEE RMPL C-9)		
A02J1421	----------			CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (SEE RMPL C-9)		
A02J1911	----------			CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (SEE RMPL C-9)		
A02J1991	----------			CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (SEE RMPL C-9)		
A02P1411	----------			CONN,BOX:SHUNT/SHORTING,;FEMALE,1 X 2,0.1 (SEE RMPL C-9)		
A02P1412	----------			CONN,BOX:SHUNT/SHORTING,;FEMALE,1 X 2,0.1 (SEE RMPL C-9)		
A02P1421	----------			CONN,BOX:SHUNT/SHORTING,;FEMALE,1 X 2,0.1 (SEE RMPL C-9)		
A02U1831	----------			SOCKET,PGA:PCB,R4000MC;447 POS,39 X 39,0.05 (SEE RMPL C-9)		

Replaceable Mechanical Parts

This section contains a list of the components that are replaceable for the 92DM75A Probe Adapter. As described below, use this list to identify and order replacement parts.

Parts Ordering Information

Replacement parts are available from or through your local Tektronix, Inc. service center or representative.

Changes to Tektronix instruments are sometimes made to accommodate improved components as they become available and to give you the benefit of the latest circuit improvements. Therefore, when ordering parts, it is important to include the following information in your order:

- Part Number
- Instrument Type or Model Number
- Instrument Serial Number
- Instrument Modification Number, if applicable

If a part you order has been replaced with a different or improved part, your local Tektronix service center or representative will contact you concerning any change in the part number.

Change information, if any, is located at the rear of this manual.

Module Replacement

The 92DM75A Probe Adapter is serviced by module replacement so there are three options you should consider:

Module Exchange — In some cases you may exchange your module for a remanufactured module. These modules cost significantly less than new modules and meet the same factory specifications. For more information about the module exchange program, call 1-800-TEKWIDE, extension BV 5799.

Module Repair — You may ship your module to us for repair, after which we will return it to you.

New Modules — You may purchase new replacement modules in the same way as other replacement parts.

Using the Replaceable Parts List

The tabular information in the Replaceable Parts List is arranged for quick retrieval. Understanding the structure and features of the list will help you find the all the information you need for ordering replacement parts.

Item Names

In the Replaceable Parts List, an Item Name is separated from the description by a colon (:). Because of space limitations, an Item Name may sometimes appear as incomplete. For further Item Name identification, U.S. Federal Cataloging Handbook H6-1 can be used where possible.

Indentation System

This parts list is indented to show the relationship between items. The following example is of the indentation system used in the Description column:

1 2 3 4 5 Name & Description
Assembly and/or Component
Attaching parts for Assembly and/or Component
(END ATTACHING PARTS)
Detail Part of Assembly and/or Component
Attaching parts for Detail Part
(END ATTACHING PARTS)
Parts of Detail Part
Attaching parts for Parts of Detail Part
(END ATTACHING PARTS)

Attaching Parts always appear in the same indentation as the item it mounts, while the detail parts are indented to the right. Indented items are part of, and included with, the next higher indentation. Attaching parts must be purchased separately, unless otherwise specified.

Abbreviations

Abbreviations conform to American National Standards Institute (ANSI) standard Y1.1.

CROSS INDEX – MFR. CODE NUMBER TO MANUFACTURER

Mfr. Code	Manufacturer	Address	City, State, Zip Code
00779	AMP INC	2800 FULLING MILL PO BOX 3608	HARRISBURG PA 17105
26742	METHODE ELECTRONICS INC	7447 W WILSON AVE	CHICAGO IL 60656–4548
58050	TEKA PRODUCTS INC	45 SALEM ST	PROVIDENCE RI 02907
63058	MCKENZIE TECHNOLOGY	44370 OLD WARMS SPRINGS BLVD	FREMONT CA 94538
80009	TEKTRONIX INC	14150 SW KARL BRAUN DR PO BOX 500	BEAVERTON OR 97077–0001

Fig. & Index No.	Tektronix Part No.	Serial No. Effective	Serial No. Dscont	Qty	12345 Name & Description	Mfr. Code	Mfr. Part No.
C–8	010–0555–00			1	PROBE ADAPTER:R4000PC,PGA179,SOCKETED	80009	010055500
–1	671–2652–00			1	CIRCUIT BD ASSY:R4000PC,PROBE ADAPTER	80009	671265200
–2	131–5267–00			4	CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (J1101,J1181,J1412,J1421,J1911,J1962,J1991 QUANTITY 8 EA)	80009	131526700
–3	131–1857–00			1	CONN,HDR:PCB,;MALE,STR,1 X 36,0.1 CTR (J1281 QUANTITY 16 EA)	58050	082–3644–SS10
–4	131–4530–00			1	CONN,HDR:PCB,;MALE,STR,1 X 3,0.1 CTR,0.230 (J1411 QUANTITY 3 EA)	00779	104344–1
–5	131–4356–00			1	CONN,BOX:SHUNT/SHORTING,;FEMALE,1 X 2 (P1411,P1412,P1421)	26742	9618–302–50
–6	136–1203–00			2	SOCKET,PGA:PCB,R4000;179 POS,18 X 18,0.1 CT (U1831)	63058	PGA179H101B1–18
					STANDARD ACCESSORIES		
	070–8750–00			1	MANUAL,TECH:INSTR,92DM75A,R4000PC/SC/MC	80009	070875000
	061–3960–00			1	SHEET,TECHNICAL:R4000PC/SC/MC,REL NOTES	80009	061396000
	063–1398–00			1	SOFTWARE PKG:R4000PC/SC/MC,SUPPORT	80009	063139800

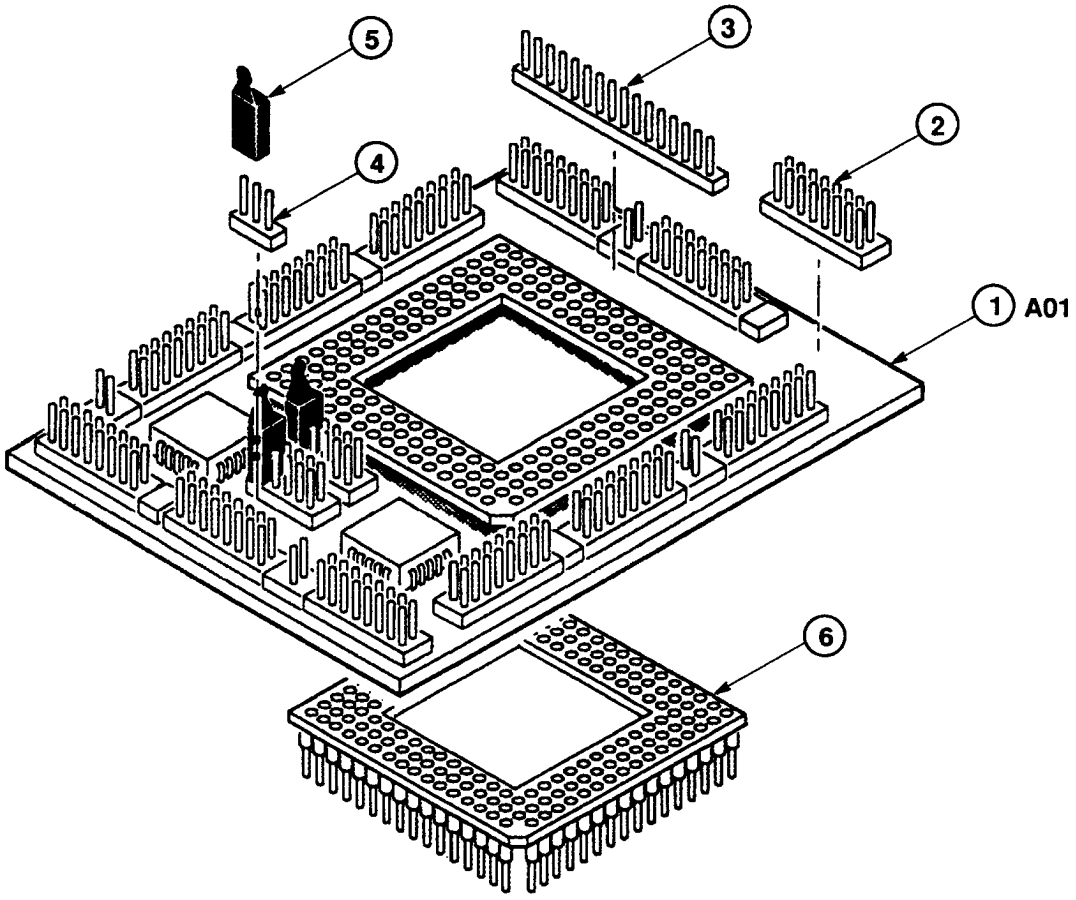


Figure C-8. Standard probe adapter exploded view.

Fig. & Index No.	Tektronix Part No.	Serial No. Effective	Dscont	Qty	12345 Name & Description	Mfr. Code	Mfr. Part No.
					OPTION 2S		
C-9	010-0562-00			1	PROBE ADAPTER:R4000SC/MC,PGA447	80009	010056200
-1	671-2772-00			1	CIRCUIT BD ASSY:R4000SC/MC,PROBE ADAPT.	80009	671277200
-2	131-5267-00			3	CONN,HDR:PCB,;MALE,STR,2 X 40,0.1 CTR,0.235 (SEE REPL A02 J1101,J1181, J1412, J1421, J1911, J1991J1280, J1281)	80009	131526700
-3	131-4857-00			2	CONN,PLUG,ELEC:200/300 FC (SEE REPL A02 J1280, J1281)	80009	131485700
-4	131-4530-00			1	CONN,HDR:PCB,;MALE,STR,1 X 3,0.1 CTR,0.230 (SEE REPL J1411)	00779	104344-1
-5	131-4356-00			3	CONN,BOX:SHUNT/SHORTING,;FEMALE,1 X 2,0.1 (P1411,P1412,P1421)	26742	9618-302-50
-6	136-1205-00			2	SOCKET,PGA:PCB,R4000MC;447 POS,39 X 39 (U1831)	63058	PZA447H120B5-39
					STANDARD ACCESSORIES		
	070-8750-00			1	MANUAL,TECH:INSTR,92DM75A,R4000PC/SC/MC	80009	070875000
	061-3960-00			1	SHEET,TECHNICAL:R4000PC/SC/MC,REL NOTES	80009	061396000
	063-1398-00			1	SOFTWARE PKG:R4000PC/SC/MC,SUPPORT	80009	063139800

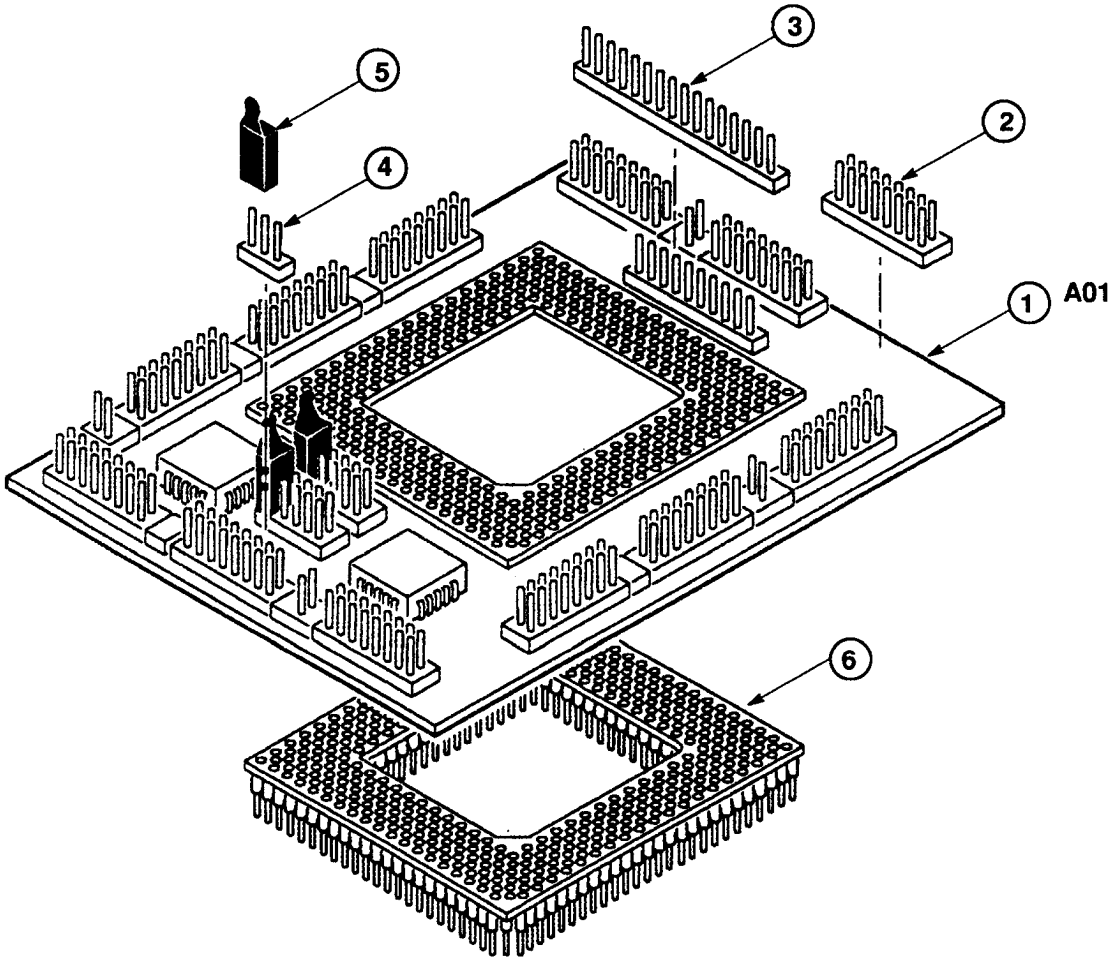


Figure C-9. Option 2S probe adapter exploded view.

INDEX

A

B

C

D

E

F

G

H

I

J

M

N

O

P

R

S

T

U

V