intel®

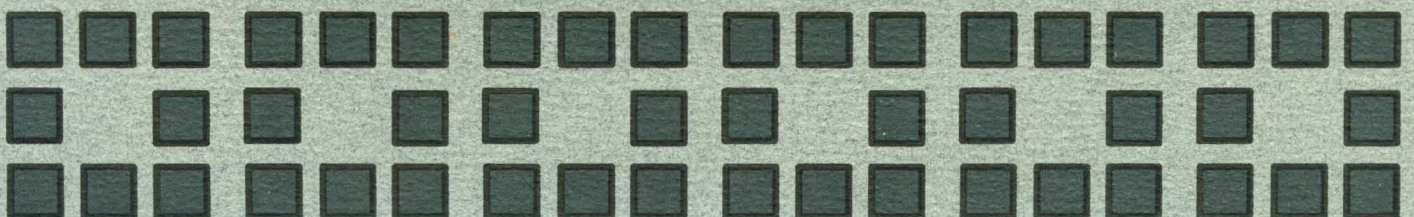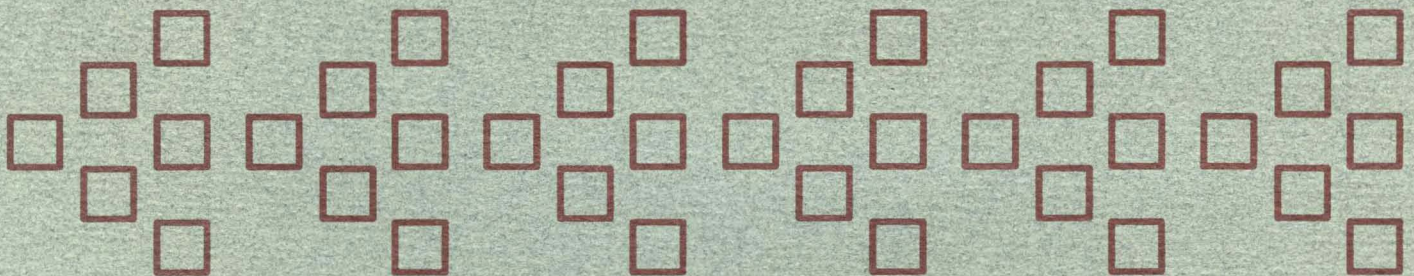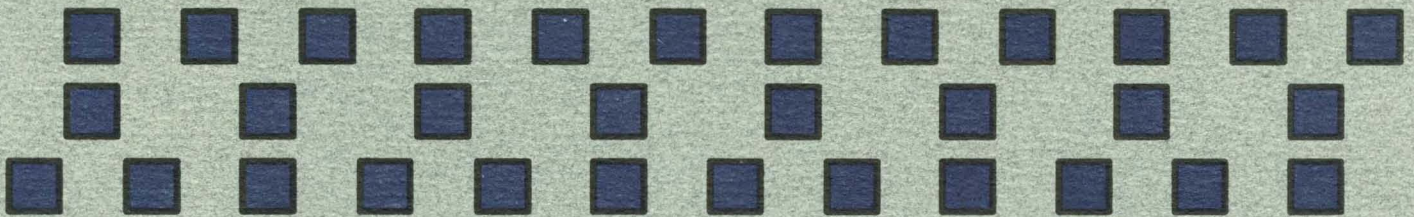# iRMX® 286/iRMX® II Troubleshooting Guide

Q1/1990

Order Number: 273472-001

intel®

iRMX® 286/iRMX® II Troubleshooting Guide

Q1/1990

This publication is produced by Intel's Technical Support Operation. Its circulation is restricted to customers who have purchased a Software Support Contract and to Intel employees who need this information.

Technical Support Operation produces Troubleshooting Guides for the following environments:

Ada386
COMMUNICATIONS iNA and Networking Products
PC-DOS Systems
iRMK Real-Time Kernel
iRMX 86 Systems
iRMX II Systems
Series II/800
Series III/IV
VAX Cross-Software
XENIX Systems

The Software Support TIPS number is 1-800-INTEL-4-U for U.S. customers only. European customers should contact their local sales office (see below).

The following problem lists have been compiled to assist you in identifying solutions to known product anomalies. To use this list, locate the symptom displayed by your software. Then, read the cause of your problem and implement the solution.

If you detect a problem with your software, please submit a Software Problem Report (SPR) to Intel Corporation. We ask that you do this even if you believe you have encountered a known problem. We can then verify your diagnosis and obtain a measure of the problem's impact on our customers.

EUROPEAN SOFTWARE SUPPORT TECHNICAL INFORMATION PHONE SERVICE

The following list shows the software support addresses and telephone numbers for each of the European sales areas:

UNITED KINGDOM

Intel Corporation (UK) Ltd.
Pipers Way
Swindon SN3 1RJ
Tele: (0793) 641469

GERMANY

Intel Semiconductor GmbH
Dornacher Strasse 1
Postfach 213
D-8016
Feldkirchen bei Munchen
Tele: 089-9032025 (software)
      089-9038529 (hardware)

SWITZERLAND & AUSTRIA

Call Hotline in Germany

FRANCE	Intel Corporation 1 Rue Edison - BP 303 78054 St. Quentin-en-Yvelines Cedex Tele: 30-57-70-17 or 30-64-56-95
SPAIN & PORTUGAL	Zurbaran 28-1-Izqda 28010 Madrid SPAIN Tele: 1-3082552
NETHERLANDS	Intel Benelux Alexander Poort Building Marten Meesweg 93 3068 AV Rotterdam Tele: 010-4071111
BELGIUM	Intel Corporation S.A. Avenue Winston Churchill 11 1180 Brussels Tele: 02-2452323
SWEDEN	Intel Sweden AB Dalvgen 24 171 36 Solna Tele: 08-7340100
DENMARK & NORWAY	Call Hotline in Sweden
FINLAND	Intel Finland OY Ruosilantie 2 00390 Helsinki 39 Tele: 90-544644
ITALY	Intel Corporation Italia Spa Milanofiori Palazzo E 20090 Assago Milano Tele: 02-89200950
ISRAEL	Intel Semiconductor Ltd. Attidim Indusrial Park P.O. Box 43202 Tel Aviv 61430 Tele: 03-498080

CONTENTS

CHAPTER 1. INTRODUCTION

CHAPTER 2. iRMX® 286 OPERATING SYSTEM Release 2.0   I

CONTENTS

CONTENTS

CHAPTER 3. iRMX® II OPERATING SYSTEM R3.0

CONTENTS

CHAPTER 4. iRMX® II OPERATING SYSTEM R4.0

CONTENTS

CONTENTS

CHAPTER 5. LANGUAGES - ASM286 V1.3

CHAPTER 6. LANGUAGES - ASM86 V3.2

CHAPTER 7. LANGUAGES - BLD286 V3.2

CHAPTER 8. LANGUAGES - BND286 V3.1

CHAPTER 9. LANGUAGES - BND286 V3.2

CHAPTER 10. LANGUAGES - iC-286 V3.2

CHAPTER 11. LANGUAGES - iC-286 V4.1

CONTENTS

CHAPTER 17. LANGUAGES - LOC86 V2.5

CHAPTER 18. LANGUAGES - MAP286 V3.2

CHAPTER 19. LANGUAGES - OVL286 V3.2

CHAPTER 20. LANGUAGES - OH86 V1.0

CHAPTER 21. LANGUAGES - PASCAL 286 V3.2

CONTENTS

CHAPTER 24. TOOLS AND UTILITIES - AEDIT 286 V2.1

CHAPTER 25. TOOLS AND UTILITIES - NUMERICS V3.0

CHAPTER 26. TOOLS AND UTILITIES - 8087 NUMERICS SUPPORT

CHAPTER 27. TOOLS AND UTILITIES - 80287 NUMERICS SUPPORT

CHAPTER 28. TOOLS AND UTILITIES - Soft-Scope* 286 DEBUGGER V2.55

APPENDIX A. SOFTWARE PRODUCT CURRENT VERSION LISTS

INTRODUCTION 1

This chapter introduces the iRMX 286/iRMX II Systems Troubleshooting Guide. The chapters in this publication have been renumbered to accommodate the purging of obsolete products and the addition of new versions of products.

The structure of this publication is as follows:

USING THE TROUBLESHOOTING GUIDE

The Table of Contents, located in the front of the publication, is the key to the entire publication. Each environment has its own chapter and each problem, or bug, is listed within its corresponding chapter number. The page numbers also correspond to the chapter number (i.e., Chapter 2, page 2-n, 2-n).

The information concerning a problem is still organized in a SYMPTOM, CAUSE, SOLUTION format. Any problems, or bugs, added to an environment or changed since the last publication, are identified by change bars.

DISTRIBUTION PROCEDURE

The entire troubleshooting guide is being provided for Q1, 1990. The 1989 publication may be discarded. Only updated chapters wil be published and distributed for the Q2, Q3, and Q4 editions.

The Request For Reader's Comments form is still included in the back of the publication. Please use this post-paid form if you have comments or suggestions regarding this publication.

iRMX® 286 OPERATING SYSTEM Release 2.0

2

This chapter lists known problems related to the iRMX 286 Operating System, Release 2.0. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, are identified by change bars.

This product is no longer supported. This chapter will be removed from the next troubleshooting guide revision.

The structure of the chapter is as follows:

2.1 SYSTEM BOOTING PROBLEMS

2.1.1 Installation System Does Not Boot

SYMPTOM The system hangs right after the sign-on message, or never comes up, while installing the system.

CAUSE This is a problem with the iSBC™ 215G firmware Version 2.3 (Part Number 146276-001 and 146276-002).

SOLUTION Use iSBC™ 215G firmware Version 2.4 (Part Number 146979-001 and 146979-002) or more current versions. Contact your local sales office for assistance.

CAUSE You do not have the right stepping of the iAPX 286 processor.

SOLUTION Use at least a B-2 stepping iAPX 286 processor. Contact your local sales office for assistance.

2.1.2 System Will Not Reboot With iSBC 215G Controller

SYMPTOM A system that worked previously now fails to initialize when an iSBC 215G controller was added to the system.

CAUSE The iSBC 215G controller uses a wakeup address in memory that corresponds to the wakeup port address jumpered on the iSBC 215G board. This address may be inadvertently written over by the system.

SOLUTION Be sure your wakeup address is not included in the memory specified for the system or for the free space manager by the memory screen in the ICU. For example, the default wakeup port address for the iSBC 215G board is 100H, the memory address used is 1000H through 1100H and must be excluded from the available memory in the ICU.

2.1.3 System With Multiple iSBC 215G Boards Hangs

SYMPTOM You have two or more iSBC™ 215G boards and when you attach a disk, the system hangs.

CAUSE The wakeup port addresses are not a minimum of 11H apart so the 272 bytes required for each board have not been reserved.

SOLUTION If the first board has wakeup port address of 100H the second one should have 111H and the third one should have 122H etc.

2.1.4 The System Fails To Boot When Using The 8251A Device Driver

SYMPTOM The system does not boot when the /config/terminals file assigns a Human Interface user to a terminal attached to an iSBX 351 module.

CAUSE To be determined.

SOLUTION Mount the iSBX 351 board or, if the iSBX 351 module is not mounted, remove the terminal attached to the iSBX 351 module from the /config/terminals file. This problem will be fixed in a future update.

2.1.5 Using The iSBC xxxCX iLBX™ RAM Boards

SYMPTOM   If your system contains more than 896 Kbytes of physical memory and you are using an iSBC xxxCX RAM Board (e.g., iSBC™ 010CX, iSBC™ 012CX, etc.), the system hangs.

CAUSE   The starting address of the Human Interface user partition lies near the 896 Kbyte boundary.

SOLUTION   You must configure your system so that no user partition begins within 50 Kbytes of the 896 Kbyte boundary.

2.1.6 Baud Rate Scan Of Terminal Set To 9600 Fails

SYMPTOM   A baud rate scan of a terminal set to 9600 baud causes the system to fail, if your system contains either an iSBC board with clockrate greater than 6MHz (such as the iSBC 286/10A board or the iSBC 286/12 board) or an iSBC board that utilizes 1 or 0 wait-state memory and uses an 8274 Terminal Driver.

CAUSE   Timing problem in interfacing with the 8274 MPC.

SOLUTION   Use two or more wait-states in memory or set the terminal baud rate to something other than 9600 baud when doing the baud rate scan or configure the 8274 Terminal Driver to use a fixed baud rate.

2.1.7 "no-go" From Winchester

SYMPTOM   When an iRMX system box is powered up, the SCT shows a "no-go" for the Winchester disk, but subsequent boots while system is powered up show a "go" for the disk.

CAUSE   If your system has "autolock" for the disk, the SCT will fail on power up for the disk because the delay is not long enough to allow for the auto-unlock. The term "autolock" means that when the power is turned off, the heads are automatically locked, versus manual lock where a switch had to be manually set to lock the heads.

SOLUTION   A future release of the SCT will have a longer delay.

2.1.8 No Response To Keyboard - No Output To Console

SYMPTOM   When you attempt to boot the system, it accesses the disk, but neither responds to keyboard input nor sends any output to the console.

CAUSE   The third stage Bootstrap Loader is in the same memory that is reserved for the second stage of the Bootstrap Loader.

SOLUTION   If you configure a third stage Bootstrap Loader, do not locate it in the space allocated for the second stage of the Bootstrap Loader. See table.

SCT Version Number	Starting RAM Location	Size of RAM Area (in bytes)
SCT 286/300x (all versions)	24K bytes less than the largest RAM address (not to exceed 0C0000H)	21K (5400H)

CAUSE   The initial task belonging to a first-level job has a missing or unexecuted END$INIT$TASK system call.

SOLUTION   Check that the initial task of all first-level user jobs invokes an END$INIT$TASK system call.

CAUSE   The :CONFIG:TERMINALS file has been corrupted, or has zero users allocated.

SOLUTION   Create or correct the :CONFIG:TERMINALS file.

CAUSE   The :CONFIG:USER/<user id> files are incorrect. A common error is to use both a comma (,) and a carriage return as a parameter delimiter. If a comma is followed by a carriage return, a null parameter is assigned.

SOLUTION   Correct the :CONFIG:USER/<user id> files to ensure that commas are not unintentionally followed by carriage returns.

CAUSE   Insufficient memory to complete the booting process.

SOLUTION   Check that your system has enough memory to complete booting.

CAUSE   Memory-mapped I/O addresses were not excluded from the SYS and FSM memory.

SOLUTION   Exclude memory-mapped I/O addresses from iRMX 286 memory using the MEMORY FOR SYSTEM screen of the ICU.

CAUSE   The system has been configured to use the System Debug Monitor (iSDM™), but the iSDM monitor has not been physically installed in the system or it has been placed at an invalid address.

SOLUTION   Install the iSDM monitor. For information about how to install and/or use the iSDM monitor, refer to the *iSDM™ System Debug Monitor Installation and Configuration Guide*.

CAUSE   The system device name in the Bootstrap Loader does not have a corresponding device name in a Device Unit Information entry within the system definition file.

SOLUTION   Using the ICU, make sure that the device name of the booting device exists in a Device Unit Information entry. For example, "W0" for the Winchester disk.

CAUSE   There is a conflict over an interrupt. For example, two or more interrupts are assigned to the same level.

SOLUTION   Check the board jumpers and the configuration definition file for two interrupts assigned to the same level.

CAUSE   The interrupt from a device is not getting to the 8259A Programmable Interrupt Controller.

NOTE

The Bootstrap Loader does not use interrupts.

SOLUTION   Check that the interrupt is actually getting to the 8259A (PIC). Refer to the hardware reference manual for your Single Board Computer (iSBC).

CAUSE   The disk has been formatted using the iRMX 86 Release 7 FORMAT and the second stage of the bootstrap loader for the iRMX 86 Release 7 system is not compatible with the iRMX 286 system.

SOLUTION   Format the disk using iRMX 286 FORMAT.

2.1.9 ROM System Does Not Come Up

SYMPTOM When a ROM based iRMX 286 operating system is powered up, it does not come up.

CAUSE The iSBC 286/10A or iSBC 286/10 board is not jumpered correctly.

SOLUTION To specify four 27512 EPROM devices, set up jumpers 62 through 91 as specified in the 27512 (64K x 8 EPROM) example in Appendix E of the *iSBC 286/10A or iSBC 286/10 Hardware Reference Manual.*

To specify a starting memory address and memory size for local memory, use primary decode option 3. The jumpers required are:

Default Configuration	Jumpers for Primary Decode Option 3
E218 - E219 installed	E218 - E219 installed
E220 - E221 removed	E220 - E221 installed

To specify memory size/justification for local memory, use secondary option 3. The jumpers required are:

Default Configuration	Jumpers for Primary Decode Option 3
E51 - E59 removed	E51 - E59 removed
E50 - E58 removed	E50 - E58 installed
E49 - E57 installed	E49 - E57 installed

2.1.10 EIOS Initialization Error 40h Encountered

SYMPTOM EIOS Initialization Error 40h is encountered when bootstrap loading an iRMX 286 operating system.

CAUSE The cause is under investigation.

SOLUTION Re-boot your iRMX 286 operating system. Will be fixed in a future release.

2.1.11 Error 118 Received From BLD286 Utility

SYMPTOM The BLD286 utility generates the following error message during system configuration.

```
*** ERROR 118: Input Segments Exceed Target Memory Processing Aborted
```

CAUSE The amount of memory needed for the iRMX 286 operating system code and data segments has exceeded the amount of memory given to it during system configuration.

SOLUTION Use the ICU286 utility and increase the system memory.

2.1.12 Checksum Error Encountered While Bootloading The Operating System

SYMPTOM A checksum error is encountered when you bootload the iRMX 286 operating system.

CAUSE A file was specified in the first stage of the bootstrap loader that was not the third stage of the bootstrap loader.

SOLUTION Always specify the third stage of the bootstrap loader when bootloading your system.

2.2 HARDWARE EXCEPTION PROBLEMS

2.2.1 General Protection In A System Call

SYMPTOM You get a general protection error within a system call, even though the parameters passed to the call are correct.

CAUSE The application program is bound to the wrong interface library. For example, a LARGE program is bound to a COMPACT interface library.

SOLUTION Check your program size control statement and the bind command to make sure they are compatible.

2.2.2 General Protection In An Application Program

SYMPTOM When running an application program, you get a general protection error.

CAUSE You have an invalid data segment or you have crossed the boundary of the segment.

SOLUTION Make the necessary correction to your program.

NOTE

The iSDM command "ddt(ds)" will display the data segment characteristics. By examining the data segment you can identify the problem.

CAUSE You have an invalid extra segment or you have crossed the boundary of the segment.

SOLUTION Make the necessary correction to your program.

NOTE

The iSDM command "ddt(es)" will display the extra segment characteristics. By examining the extra segment you can identify the problem.

2.2.3 Stack Protection In An Application Program

SYMPTOM When running an application program you get a stack protection error.

CAUSE You have an invalid stack segment or the stack has overflowed.

SOLUTION Make the necessary correction to your program.

NOTE

The iSDM command "ddt(ss)" will display the stack segment characteristics. By examining the stack segment you can identify the problem.

CAUSE The PL/M-286 compiler did not create a stack and the application loader did not create a stack big enough for the execution of the program.

SOLUTION Use debug control when compiling.

2.2.4 Stack Protection Error When Running BLD286

SYMPTOM When running BLD286 on a system that does not have enough disk space, a stack protection exception is encountered. The builder displays the following messages a number of times before it gets a protection error:

```
INTERFACE ERROR
0029 : E$SPACE
FILE : ROM.MP2
PROCESS ABORTED
```

CAUSE The disk does not have enough space to contain the output files from the build operation.

SOLUTION Make sure your disk has over 1 Mbyte free space before starting the ICU. This problem will be fixed in a future release of BLD286.

2.2.5 Third Stage Gets DIVIDE-BY-ZERO Error

SYMPTOM The third stage of the Bootstrap Loader gets a DIVIDE-BY-ZERO error.

CAUSE Your system was not reset before you booted your iRMX 286 operating system.

SOLUTION Reset the system by pressing the reset button on your system's front panel or by whatever means you have built into your hardware.

CAUSE You have configured the IDT with too few slots.

SOLUTION Check your definition file and correct the number of IDT entries in the Nucleus screen of the ICU.

CAUSE You are using an iRMX 86 operating system Bootstrap Loader second stage. In other words, you are booting from a disk which was formatted using the iRMX 86 FORMAT command.

SOLUTION Reformat the disk using iRMX 286 FORMAT command with BOOTSTRAP option.

2.3   APPLICATION LOADER PROBLEMS

2.3.1   Wrong Message Returned To MSG$MBOX In RQ$S$LOAD$IO$JOB

SYMPTOM   When using the RQSLOADIOJOB system call, the mailbox specified in the MSG$MBOX parameter receives the wrong message.

CAUSE   The mailbox specified in the MSG$MBOX parameter of the RQ$S$LOAD$IO$JOB call is used internally by the loader to receive the loader result segment from RQALOADIOJOB. This causes several problems:

1. The user cannot specify selector$of(NIL) for the MSG$MBOX parameter as in RQ$CREATE$IO$JOB.

2. The user can receive the wrong message. If the user has a separate task to receive exit messages from I/O jobs and this task does a RQ$RECEIVE$MESSAGE at the MSG$MBOX before another task calls RQSLOADIOJOB, the user will receive the loader result segment and the system will crash.

3. The user can receive the wrong message. If the user has 2 RQSLOADIOJOB calls that specify the same MSG$MBOX, the loader can receive the exit message from the first I/O job while expecting to get the loader result segment from the second load. This prevents delivery of the exit message and causes the second load to terminate without success.

SOLUTION   As a workaround, avoid the following:

1. Specify NIL for the MSG$MBOX of the RQ$S$LOAD$IO$JOB

2. Leave a task waiting for a message at a mailbox and then use the same mailbox in a RQSLOADIOJOB call.

This problem will be fixed in a future update or release.

2.3.2   General Protection Error When Running Program

SYMPTOM   When trying to run a large program you encounter a General Protection error.

CAUSE   An error in the Application Loader.

SOLUTION   Reduce the number of fix-up records in the file to be loaded by reducing the amount of POINTER data types used in your program.

2.4   BOOTSTRAP LOADER PROBLEMS

2.4.1   Third Stage Does Not Issue The First Message

SYMPTOM   While bootloading, the message, NOW IN PROTECTED MODE, is not issued and the system hangs.

CAUSE   The load file you specified with a .286 extension or the default load file you used, /SYSTEM/RMX86.286 (if no name is specified) was not found on the volume.

SOLUTION   Check the volume to make sure the file specified or the default file /SYSTEM/ RMX86.286 exists.

CAUSE   The pathname for the third stage has a period in it other than the period for the file name extension.

SOLUTION   Rename the directory or the filename in the pathname of the third stage, so that the name does not contain a period, except the period for the extension of the third stage's file name. For example, if the third stage has a pathname /BOOT/MY.DIR/SYS286.3RD, change it to /BOOT/MYDIR/SYS286.3RD.

2.4.2 Third Stage Does Not Issue The Second Message

SYMPTOM   While bootloading, the message, TARGET FILE LOADED, is not issued and the system hangs.

CAUSE   The file containing the system was incorrect.

SOLUTION   Check the file and regenerate the system if necessary.

CAUSE   The third stage was loading the load file on top of it.

SOLUTION   Use the memory screen of the ICU to correct your memory allocation. By default, the generic third stage is located at B8000H - C0000H, the device specific third stage is located at BC000H - C0000H. Your system memory should not include this location. However, the free space manager can include this memory.

2.4.3 Third Stage Gets A DIVIDE-BY-ZERO Error

SYMPTOM   The third stage of the Bootstrap Loader gets a DIVIDE-BY-ZERO error.

CAUSE   Your system was not reset before you booted your iRMX 286 operating system.

SOLUTION   Reset the system by pressing the reset button on your system's front panel or by whatever means you have built into your hardware.

CAUSE   You have configured the IDT with too few slots.

SOLUTION   Check your definition file and correct the number of IDT entries in the Nucleus screen of the ICU.

CAUSE   You are using an iRMX 86 operating system Bootstrap Loader second stage. In other words, you are booting from a disk which was formatted using the iRMX 86 FORMAT command.

SOLUTION   Reformat the disk using the iRMX 286 FORMAT command with the BOOTSTRAP option.

2.4.4 Checksum Error Encountered While Bootloading The Operating System

SYMPTOM   A checksum error is encountered when you bootload the iRMX 286 operating system.

CAUSE   A file was specified in the first stage of the bootstrap loader that was not the third stage of the bootstrap loader.

SOLUTION   Always specify the third stage of the bootstrap loader when bootloading your system.

2.5   I/O SYSTEM, DEVICE, AND DRIVER PROBLEMS

2.5.1   System Time Not Current When CONTROL-C Out Of TIME Command

SYMPTOM   If you abort the default r?logon program at signon time via a CONTROL-C, the default Date/Time of the system does not automatically revert to the last time the directory :SYSTEM: was accessed.

CAUSE   The system gets its time from the :SYSTEM: directory if the operator does not enter time during signon time. To improve performance, iRMX 286 does not update a file's access parameter unless the file was modified. Because of this, the access time of :SYSTEM: directory reflects the last time it was modified.

SOLUTION   Enter time each time you logon. Run "SHUTDOWN" before you turn power off or reset your system.

2.5.2   iSBC 188/48 Board Ignores Characters With Parity Errors

SYMPTOM   The iSBC 188/48 driver does not pass thru characters with input parity errors.

CAUSE   A feature of the iSBC 188/48 firmware.

SOLUTION   Use a different device. For example, an iSBC 544A board may be used.

2.5.3   iSBC 188/48 and iSBC 188/56 Board Problem

SYMPTOM   A General Protection error is encountered in a system that uses the iSBC 188/48 or iSBC 188/56 board.

CAUSE   A problem with the iSBC 188/48 and iSBC 188/56 boards (if IC U-69 is not labeled 451012).

SOLUTION   Return the board to the repair center for upgrade.

2.5.4   Line Printer Hangs System

SYMPTOM   The line printer hangs the system when the printer is OFFLINE and then turned back ONLINE.

CAUSE   There is a problem with ZAP's ZAXAB2 and ZAXAB3.

SOLUTION   Enter a CONTROL-C to stop the line printer from hanging the port. Do not apply the ZAPs or make sure your printer is ONLINE when you copy a file to it. This problem will be fixed in a future update or release.

2.5.5   Translation Of Escape Sequence #18 Does Not Work

SYMPTOM   Translation of escape sequence 18 does not work.

CAUSE   A problem in the Terminal Support Code.

SOLUTION   This problem will be fixed in a future update or release.

2.5.6 A TCC Channel Appears To Hang

SYMPTOM   A terminal on a TCC(iSBC 188/48, iSBC 188/56, iSBC 546, iSBC 547, iSBC 548) port appears to hang.

CAUSE   A problem with the TCC driver.

SOLUTION   This problem is fixed in Update 1 by ZAP ZXA015.

2.5.7 Terminal Port Hangs On Output

SYMPTOM   Single character writes to a terminal hangs the terminal port.

CAUSE   A problem in the Terminal Support Code.

SOLUTION   This problem is fixed in Update 1 by ZAP ZXA033.

2.5.8 Type Ahead Causes Terminal To Hang

SYMPTOM   Type ahead in line edit mode hangs the terminal.

CAUSE   A problem in the Terminal Support Code.

SOLUTION   This problem is fixed in Update 0 by ZAP ZXA001

2.5.9 Terminal Device Appears To Hang

SYMPTOM   When various CONTROL-O, CONTROL-W sequences are entered on a terminal while a file is being output to the terminal, the terminal port locks up.

CAUSE   A problem in the Terminal Support Code.

SOLUTION   This problem is fixed in Update 0 by ZAP ZXA002 and ZXA003.

2.5.10 iSBC 208 Device Driver Hangs The System

SYMPTOM   When using a 5.25" floppy drive and a iSBC 208 controller the system appears to hang when the floppy is accessed.

CAUSE   A problem with the iSBC 208 driver.

SOLUTION   This problem is fixed in Update 2 by ZAPs ZXA089 and ZXA090.

2.5.11 Attaching A iSBC 534 Port Causes The iSBC 534 To Hang

SYMPTOM   The iSBC 534 driver hangs when attaching a port on the iSBC 534 board while another iSBC 534 port on the same board is inputting data.

CAUSE   An error in the iSBC 534 driver.

SOLUTION   Fixed by Class B ZAP ZXB119. For U.S. customers contact TIPS. For European customers contact your Local Sales Office.

2.5.12 CONTROL-C Does Not Work On A Human Interface User Terminal

SYMPTOM   Enter a CONTROL-C on a Human Interface user terminal and it does not abort your program.

CAUSE   An error in the terminal support code.

SOLUTION   Fixed in Update 2 by ZAPs ZXA077 thru ZXA082 and ZXA084 through ZXA086.

2.5.13 Output From A TCC Port Appears Blocked

SYMPTOM   Output from a TCC port appears to be blocked.

CAUSE   An error in the TCC driver.

SOLUTION   Fixed by Class B ZAP ZXB112. For U.S. customers contact TIPS. For European customers contact your Local Sales Office.

2.5.14 The TCC Driver Appears To Hang

SYMPTOM   The TCC appears to hang if you are operating a port at 38.4 kbaud and output flow control occurs (x-off) or if a TCC input buffer fills up while the channel is doing output.

CAUSE   An error in the TCC driver.

SOLUTION   Fixed by Class B ZAPs ZXB114 through ZXB119. For U.S. customers contact TIPS. For European customers contact your Local Sales Office.

2.5.15 Non-Buffered Terminal Port Hangs

SYMPTOM   A non-buffered terminal port (e.g., 8274) occasionally hangs.

CAUSE   An error in the Terminal Support Code.

SOLUTION   Fixed in Update 2 by ZAPs ZXA077 through ZXA086.

2.5.16 System Hangs When TCC Port Is Heavily Used

SYMPTOM   The system hangs if massive output is interspersed with input on two or more of the TCC channels.

CAUSE   An error in the TCC driver.

SOLUTION   Fixed in Update 2 by ZAP ZXA087.

2.5.17 System Hangs During Tape Operations

SYMPTOM   The system occasionally hangs when using the tape drive.

CAUSE   An error in the iSBC 214/215G/iSBX 217C/218A Driver.

SOLUTION   Fixed in Update 2 by ZAP ZXA100.

2.5.18 E$MEM Received From RQ$GET$DIRECTORY$ENTRY System Call

SYMPTOM   You get an E$MEM exception from the RQGETDIRECTORY$ENTRY system call.

CAUSE   A problem with the RQGETDIRECTORY$ENTRY system call, where it does not return 6 bytes to the available memory pool each time it is called.

SOLUTION   This problem will be fixed in a future update or release.

2.5.19 Cannot Use iSBX 251 At Addresses Other Than 80H

SYMPTOM   The iSBX 251 driver will not work if you move the iSBX 251 driver data port address, anywhere other than a data port address of 80H.

CAUSE   A problem with the ICU where the IDEVC.A28 file does not reflect the change of data port address made when configuring the driver.

SOLUTION   Edit the IDEVC.A28 file to reflect the correct data port address. This problem will be fixed in a future update or release.

2.5.20 Establishing iRMX Signal Characters Causes E$LIMIT Exception

SYMPTOM   iRMX returnd an E$LIMIT exception when calling RQASPECIAL to establish a signal character. This happens even though the maximum allowable number of signal characters for this terminal has not been exceeded.

CAUSE   During a previous attach to the terminal, signal characters were established and not removed before the terminal was detached.

When the terminal is re-attached, the Terminal Support Code removes the previous signal character associations with semaphores, but fails to clear the internal semaphore storage area, resulting in a reduction of the storage space available to establish new signal characters.

SOLUTION   Ensure that any signal characters established for a terminal are removed before the terminal is detached. This will be corrected in a future update or release.

2.6   NUCLEUS PROBLEMS

2.6.1 System Is In Infinite Loop

SYMPTOM   If the SDB is not the exception handler in your system and the exception mode is "never", a DIVIDE-BY-ZERO error in your application program will cause an infinite loop to be executed.

CAUSE   The system default exception handler and the system alternative exception handler do not increment the IP register past the offending instruction.

SOLUTION   Use the SDB as the exception handler during the development phase and write a user exception handler if DIVIDE-BY-ZERO can occur in the application.

2.6.2 Cannot Set Exception Handlers For Hardware Exceptions

SYMPTOM RQSETEXCEPTION$HANDLER will not install a user-written exception handler for a hardware exception.

CAUSE The iRMX 286 Release 2.0 system does not provide this capability.

SOLUTION This problem will be fixed in a future release.

2.6.3 RQ$RESUME$TASK Call Gets E$STATE

SYMPTOM A task which makes a RQ$DISABLE$DELETION call to protect itself from deletion and then makes a RQ$SUSPEND$TASK call to suspend itself, cannot be resumed by a RQ$RESUME$TASK call. The RQ$RESUME$TASK returns a condition code of E$STATE which indicates that the task was not suspended.

CAUSE Error in NUC.LIB.

SOLUTION This problem will be fixed in a future update.

2.6.4 Exceptions Are Not Caught By The Exception Handler

SYMPTOM Environmental exceptions are not caught by the exception handler.

CAUSE The UDI exception handler acts only on programmer exceptions, and when binding your program you specified the UDI Interface Library before the iRMX 286 Interface Library.

SOLUTION In your BND86 statement specify the iRMX 286 Interface Library before the UDI Interface Library.

2.6.5 RQ$TIMED$INTERRUPT Does Not Return An E$TIME Exception

SYMPTOM If the Nucleus call, RQ$TIMED$INTERRUPT, times out, you do not recieve the E$TIME exception code, instead, you recieve E$OK.

CAUSE An error in the Nucleus.

SOLUTION Fixed by Class B ZAPs ZXB101 and ZXB102. For U.S. customers contact TIPS. For European customers contact your Local Sales Office.

2.6.6 Pending Interrupt Count Corrupted

SYMPTOM Unpredictable system behavior may result when using the Nucleus call RQ$TIMED$INTERRUPT.

CAUSE If the Nucleus system call RQE$TIMED$INTERRUPT times out, the pending interrupt count gets decremented even though no interrupt occurred.

SOLUTION Fixed by Class B ZAPs ZXB101 and ZXB102. For U.S. customers contact TIPS. For European customers contact your Local Sales Office.

2.6.7 E$PARAM Exception Is Not Returned From System Calls

SYMPTOM   The E$PARAM exception code is not returned from iRMX system calls when the call is passed bad parameters.

CAUSE   An error in the Nucleus. The child task was not getting the parent job validation parameter.

SOLUTION   Fixed by Class B ZAPs ZXB120, ZXB121, and ZXB122. For U.S. customers contact TIPS. For European customers contact your Local Sales Office.

2.6.8 E$STATE Exception Never Returned By RQ$RESUME$TASK

SYMPTOM   An E$STATE exception is not returned if you resume a non-suspended task with the Nucleus call RQ$RESUME$TASK.

CAUSE   An error in the Nucleus.

SOLUTION   Will be fixed in a future release.

2.6.9 Problem With The RQ$RESUME$TASK() System Call

SYMPTOM   If you resume a non-suspended sleeping task with the nucleus system call RQ$RESUME$TASK, the nucleus does not return an E$STATE exception.

CAUSE   Problem with the RQ$RESUME$TASK system call.

SOLUTION   This problem will be fixed in a future update or release.

2.6.10 RQEOBJECT$ACCESS Returns An E$BAD$ADDRESS Exception

SYMPTOM   The RQEOBJECT$ACCESS nucleus system call returns a EBADADDRESS exception if the ACCESS$PTR parameter of this call points to a structure that is located at the end of the data segment.

CAUSE   Problem with the RQEOBJECT$ACCESS system call.

SOLUTION   This problem will be fixed in a future update or release.

2.7 UNIVERSAL DEVELOPMENT INTERFACE (UDI) PROBLEMS

2.7.1 CONTROL-C Handler Gets Stack Protection Error

SYMPTOM   A user-written CONTROL-C handler gets a Stack Protection error.

CAUSE   The CONTROL-C handler was compiled using the COMPACT model of compilation.

SOLUTION   Compile the CONTROL-C handler using the LARGE model of compilation.

2.7.2 Run Time Error 0001H Encountered In An I/O Job

SYMPTOM   When making a FORTRAN written application an I/O job, you receive a run time exception of 0001H.

CAUSE   A problem with the UDI.

SOLUTION   Lookup "R?HCNAME" in the CLI's job and catalog it in the I/O job's directory under the same name. This problem will be fixed in a future update or release.

2.7.3 DQ$CHANGE$EXTENSION Does Not Return A E$STRING$BUFFER Exception

SYMPTOM   The UDI call DQ$CHANGE$EXCEPTION does not return an E$STRING$BUFFER exception when the filename specified in the call is larger than 14 characters.

CAUSE   A problem in the UDI.

SOLUTION   This problem is fixed in Update 1 by ZAP ZXA007.

2.8   INTERACTIVE CONFIGURATION UTILITY (ICU) PROBLEMS

2.8.1 BLD286 Issues "REFERENCE TO EMPTY SEGMENT" Error Message

SYMPTOM   If a user module contains only large model reentrant procedures and this module is specified as a user module in the ICU, then BLD286 issues an error message "REFERENCE TO EMPTY SEGMENT".

CAUSE   Reentrant procedures use the stack for their data. As a result, the data segment remains empty. In the large model, this empty data segment does not get combined with the data segments of other modules.

SOLUTION   Declare a "dummy" variable in the module at the module level, so that the data segment is not empty.

2.8.2 Changes To Definition File Were Not Saved

SYMPTOM   After generating a system, changes made in the ICU session did not get changed in the definition file.

CAUSE   The ICU tried to access files without having the correct access rights for those files.

SOLUTION   Refer to chapter two of the *iRMX™ 286 Configuration Guide* for correct procedures on access rights of ICU files.

2.8.3 UDS Does Not Work

SYMPTOM   UDS does not work.

CAUSE   A problem with UDS.

SOLUTION   This problem is fixed in Update 2 by ZAPs ZXA008 and ZXA009.

2.8.4 GDT/LDT Slots Not Available

SYMPTOM   "GDT/LDT Slots Not Available" The exception code E$SLOT is encountered when there should be plenty of GDT slots available.

CAUSE   An error in the way ICU causes the GDT/LDT slots to be improperly reserved.

SOLUTION   Fixed in Update 2 by ZAP ZXA094.

2.9 MASS STORAGE PROBLEMS

2.9.1 Cannot Format iSBX 251 Multimodule

SYMPTOM   The user cannot format an iSBX 251 Magnetic Bubble Memory Multimodule.

CAUSE   Improper jumpering on iSBX 251 multimodule.

SOLUTION   On the iSBX 251 multimodule, remove jumper W5 and install jumper W6. For more information on jumpering the iSBX 251 multimodule, refer to the *iSBX™ 251 Multimodule Technical Reference Manual.*

2.9.2 Data Loss On iSBX 251 Driver

SYMPTOM   Data loss occurs on an iSBX 251 driver.

CAUSE   The CPU must service all interrupts from the iSBX 251 driver within 1.5 milliseconds. Failure to respond within this time frame could cause the iSBX 251 driver to lose data.

SOLUTION   Solutions are as follows:

1. Do not mount the iSBX 251 Multimodule on a board with a clockrate greater than 6MHz (such as an iSBC 286/12).

2. Do not mount the iSBX 251 Multimodule on a CPU board that uses less than two wait-states on the iLBX.

3. Do not use a terminal that uses the terminal support code.

2.9.3 Data Loss On Mass Storage Device

SYMPTOM   "Data Loss on Mass Storage Device" Data loss occurs on a mass storage device.

CAUSE   System data buffers have not been written to the storage device prior to power loss or pressing the reset button.

SOLUTION   Following disk access, wait for the update timeout (usually two seconds) before pressing the reset button or powering down the system.

NOTE

It is good practice to close all files and use the system call A$UPDATE while accessing disk files under program control. When rebooting or powering down your system, always run shutdown first.

2.9.4 Data Loss On A Diskette

SYMPTOM   Data loss occurs on a diskette.

CAUSE   Some diskette drives do not have a Ready Line to inform the operating system when diskettes are changed; this applies to 5.25" drives, slim-line 8" drives, and micro floppy drives.

SOLUTION   Use the ATTACHDEVICE command after a new diskette has been inserted and use the DETACHDEVICE command before removing a diskette from the drive.

2.9.5 Cannot Use iSBX 251 At Addresses Other Than 80H

SYMPTOM The iSBX 251 driver will not work if you move the iSBX 251 driver data port address, anywhere other than a data port address of 80H.

CAUSE A problem with the ICU where the IDEVC.A28 file does not reflect the change of data port address made when configuring the driver.

SOLUTION Edit the IDEVC.A28 file to reflect the correct data port address. This problem will be fixed in a future update or release.

2.10 MULTIUSER PROBLEMS

2.10.1 Users Do Not Receive Console Output

SYMPTOM One or more users do not receive console output, and the system does not respond to keyboard input.

CAUSE The device driver for the user's terminal was configured with EIOS buffers.

SOLUTION Always set Max Buffers (MB) to zero on the Device-Unit Information screen when configuring a driver for a terminal.

CAUSE An error occurred when the system tried to bring up the users.

SOLUTION Run the INITSTATUS command from the resident user console to determine the initialization status of the users. Refer to the *iRMX™ 286 Operator's Guide To The Human Interface Reference Manual* for information on the INITSTATUS command.

CAUSE The user does not have the proper access rights to some critical directory or file.

SOLUTION Check the user's default directory (:$:), the :PROG: directory and the :CONFIG:TERMINALS file to ensure that each user has the proper access. Use the PERMIT command to change the access rights of the directory or file.

CAUSE Either the :CONFIG:TERMINALS or :CONFIG:USER/<user-id> files are incorrect.

SOLUTION Examine these two files for proper content and reboot. Refer to Chapter 20 in the *iRMX™ 286 Hardware And Software Installation Guide*.

CAUSE The resident user has been configured with a maximum memory size of 0FFFFH. Hence, the resident user takes all available memory.

SOLUTION Reset the resident user's memory requirements in the ICU. Ensure that sufficient memory is allocated to the non-resident users.

2.10.2 Memory-Related Error Messages

SYMPTOM A user receives a memory-related error message (for example, E$MEM) when trying to run utilities or the SUPER command in a multiuser system.

CAUSE Insufficient memory has been allocated for that user in the :CONFIG:TERMINALS file.

SOLUTION Refer to the Non-Resident Configuration Files Chapter of the *iRMX™ 86 Installation and Configuration Guide* for the memory partition size requirements for languages and utilities.

2.10.3 Terminals Lose Characters

SYMPTOM   In a multiuser environment, a terminal may lose characters on input.

CAUSE   The baud rate is too high.

SOLUTION   Lower the baud rate and/or reduce the number of users.

2.10.4 Stack Protection Error Occurs While Logging On

SYMPTOM   A stack protection error occurs when one user is doing massive I/O operations and another user attemts to logon.

CAUSE   An error in the Human Interface.

SOLUTION   Fixed in Update 2 by ZAP ZXA094.

2.11 DEBUGGING PROBLEMS

2.11.1 Breakpoints Ignored

SYMPTOM   Breakpoints set in ROM with the Debugger are ignored.

CAUSE   When a breakpoint is set, a software interrupt instruction is actually placed at the address specified. This instruction causes a jump to the monitor. These instructions cannot be written into ROM.

SOLUTION   Only in-circuit emulators allow stopping normal processing at a given ROM address. Either put the code in RAM for debugging purposes or use one of the in-circuit emulators.

2.11.2 VT Command Causes System To Hang

SYMPTOM   A VT command on a region object may hang the system.

CAUSE   A problem in the SDB.

SOLUTION   This problem is fixed in Update 1 by ZAP ZXA032.

2.11.3 VT Command Causes The System To Loop Infinitely

SYMPTOM   A VT on a region object may cause the system to loop infinitely while outputting the first waiting task's token to the terminal.

CAUSE   A problem in the SDB.

SOLUTION   This problem is fixed in Update 1 by ZAP ZXA032.

2.11.4 SDB Displays Bogus Objects In The Mailbox's Overflow Queue

SYMPTOM   A VT command on a mailbox object may display invalid objects on the overflow queue.

CAUSE   A problem in the SDB.

SOLUTION   This problem is fixed in Update 1 by ZAP ZXA046.

2.11.5 SDB Cannot Recognize Data Mailboxes

SYMPTOM   A VT or VO command on a mailbox object cannot distinguish between a data mailbox and message mailbox.

CAUSE   The SDB treats all mailboxes as message mailboxes.

SOLUTION   VT is fixed in Update 1 by ZAP ZXA032. VO will be fixed in a future update or release.

2.11.6 Job Flags Field Is Incorrect When Viewing A Job Token

SYMPTOM   When you view a job token with the SDB command VT, the Job Flags field is incorrect.

CAUSE   An error in the SDB.

SOLUTION   A value of 02 for Job Flags means that parameter validation is enabeled and a value of 00 means that it is disabeled. Will be fixed in a future release.

2.11.7 Problem With The SDB Command VT

SYMPTOM   If you look at a job token with the SDB command VT, an incorrect value is returned for the JOB$FLAGS field.

CAUSE   Problem with the SDB.

SOLUTION   This problem is fixed in Update 1 by class A ZAP ZXA033.RMD.

2.12   PATHNAME PROBLEMS

2.12.1 DIR Does Not List All Files

SYMPTOM   Some file names are not listed when using the Directory (DIR) command.

CAUSE   Files with "R?" or "r?" in their filenames are considered invisible. This is a feature of the operating system.

SOLUTION   Use the "Invisible" option of the DIR command to list these files, (that is, DIR :$: I).

2.12.2 Logical Name Does Not Exist

SYMPTOM   Some Human Interface commands do not function, usually generating the error "LOGICAL NAME DOES NOT EXIST".

CAUSE   Some commands assume the system disk device name is :SD:.

SOLUTION   Always configure systems with :SD: as the logical name for the system device.

2.12.3 DIR / Does Not List :SD:/

SYMPTOM   The "DIR /" command does not list the ":SD:/" directory.

CAUSE   When you change the default directory (:$:) to a different physical disk device with the ATTACHFILE command, the root directory (/) changes. The root directory (/) always refers to the root of the device being used as the default.

SOLUTION   Make specific references to :SD: to access the system device root directory.

2.12.4 Error When Attempt To Access File Or Directory

SYMPTOM   Attempting file or directory access causes errors.

CAUSE   Access has not been granted to the user ID attempting to use a file or directory.

SOLUTION   Determine the access rights with the EXTENDED option on the Directory (DIR) command. The file's owner can change the access rights by invoking the PERMIT command. Also, the access rights to directories, as well as files, should be verified.

2.12.5 Unauthorized Changes To Files Or Directories

SYMPTOM   Files and directories are being changed without the user's knowledge and/or access rights to files do not seem to work.

CAUSE   There are multiple users with the same user ID.

SOLUTION   File security can be enhanced by assigning different user IDs and default directories.

2.12.6 Filenames With A "?"

SYMPTOM   Problems arise when invoking the COPY, DELETE, or RENAME command on files with a "?" embedded in their filenames, (for example, R?LOGON).

CAUSE   A "?" is seen as a wildcard, not a filename character.

SOLUTION   Enclose this portion of the pathname in quotes, for example, COPY R"?"logon. This causes the enclosed characters to be taken literally.

2.13 HUMAN INTERFACE PROBLEMS

2.13.1 System Hangs When Attempting To Access Line Printer

SYMPTOM   An attempt to access a line printer that is either physically disconnected or is not turned on, hangs the system.

CAUSE   The driver waits forever for the interrupt from the printer.

SOLUTION   Do not use the line printer when it is disconnected or powered off. The workaround is press CONTROL-C, make the printer ready and re-issue the command.

2.13.2 System Hangs When iSBX 351 Terminal Is Turned Off

SYMPTOM   If a user using an iSBX 351 serial channel disconnects or turns off the terminal, the system will hang.

CAUSE   The iSBX 351 board bombards the iRMX 286 system with interrupts when in this condition.

SOLUTION   Do not turn off the terminal. This problem will be fixed in a future update or release.

2.13.3 Error Message "delete access required"

SYMPTOM Incorrect error message (delete access required) returned by RENAME and COPY command.

CAUSE The file's directory does not have proper access.

SOLUTION Give the file's directory delete and add permission (using the PERMIT command) for the user executing the RENAME or COPY command.

2.13.4 Cannot Boot Off A Floppy Diskette

SYMPTOM A bootable diskette has no errors from the DISKVERIFY command but the diskette will not boot.

CAUSE DISKVERIFY does not verify the bootstrap loader portion of the diskette.

SOLUTION Format the disk using the BOOTSTRAP option.

2.13.5 Unpredictable System Behavior

SYMPTOM Unpredictable system operation.

CAUSE A Human Interface command is not terminated properly. Failure to correctly terminate a command causes unknown code to be executed and may cause serious loss of data.

SOLUTION Every HI command must end with either an EIOS call "RQ_EXIT_IO_JOB" or a UDI call "DQ_EXIT". The purpose of these calls is to correctly transfer control back to the iRMX 286 Operating System.

2.13.6 RQCSENDCORESPONSE Or RQCSENDEORESPONSE

SYMPTOM Application program does not wait for response to RQCSENDCORESPONSE or RQCSENDEORESPONSE.

CAUSE The Human Interface includes the string length of the response string in the response$max parameter. But, if the response$max is one, the Human Interface returns without a response$string.

SOLUTION The value in response$max must equal the length of the string plus one.

2.13.7 Console Commands Not Executed

SYMPTOM Commands invoked from the console appear not to execute, and either the Human Interface prompt or a memory-related error message is returned.

CAUSE The size of the memory pool for the command is insufficient.

SOLUTION Increase the size of the program's memory pool in the BIND statement. Set Max to 0FFFFFH so that it can borrow memory.

CAUSE The user has insufficient memory to satisfy the memory pool requirements of the command.

SOLUTION Enlarge the memory partition size in the :CONFIG:TERMINALS file.

2.13.8 System Time Is Incorrect After A Reboot

SYMPTOM System time and date stamp are not updated.

CAUSE The BIOS does not update the access time of any file until it is modified. The Human Interface uses the last access time of the system directory (:SYSTEM:) as the default time and date.

SOLUTION Always set the time and date when you boot your system or run shutdown before resetting or powering down your system.

2.13.9 Terminal Acts Peculiar When Running RESTORE

SYMPTOM Terminal acts peculiar when running the RESTORE command.

CAUSE RESTORE sends a 012H (DC2) character to :CO: before printing out the message "Directory Restored".

SOLUTION Run RESTORE from a submit file. This problem will be fixed in a future update or release.

2.13.10 A General Protection Error Occurs When Using The ALIAS Command

SYMPTOM A General Protection error occurs when the ALIAS command is used with an alias of more than 14 characters.

CAUSE A problem in the ALIAS command.

SOLUTION This problem is fixed in Update 1 by ZAP ZXA006.

2.13.11 PASSWORD Does Not Work With Redirected Input

SYMPTOM Using the PASSWORD command with redirected input causes the terminal to hang. A CONTROL-C will abort the PASSWORD command.

CAUSE A problem in the PASSWORD command.

SOLUTION This problem is fixed in Update 1 by ZAP ZXA014.

2.13.12 Tapes That Were Good, Fail

SYMPTOM A previously functioning tape cartridge fails using the RESTORE command.

CAUSE The tape needs to be retensioned.

SOLUTION A retension command is included in Update 1 by ZAP ZXA037.

2.13.13 ERROR.LIT File Problems

SYMPTOM The ERROR.LIT file is missing the EUDFIO literal and has the wrong literal name for value 65H.

CAUSE A problem with the ERROR.LIT file.

SOLUTION This problem is fixed in Update 0 by ZAP ZXA005.

2.13.14 Output Redirection To A Line Printer

SYMPTOM   Redirecting output to a line printer by using the Command Line Interpreter (CLI) I/O redirection arrow (>) does not work.

CAUSE   A problem with the CLI.

SOLUTION   This will be fixed in a future update or release.

2.13.15 CLI I/O Redirection In An Alias

SYMPTOM   CLI I/O redirection cannot be incorporated in an alias.

CAUSE   A design limitation of the ALIAS command.

SOLUTION   Do not use CLI I/O redirection in an alias.

2.13.16 Tapes Generated On iRMX 286 R2.0 Operating System Cannot Be Read On iRMX 86 Operating System

SYMPTOM   Tapes generated on an iRMX 286 R2.0 system can not be read on an iRMX 86 R7.0 system.

CAUSE   The granularity of the tape drive for iRMX 286 R2.0 is 7e00H.

SOLUTION   Create a tape DUIB on your iRMX 86 system with a granularity of 7e00H.

2.13.17 SHUTDOWN Command Does Not Work After Installing Update 1

SYMPTOM   The SHUTDOWN command does not work after installing Update 1.

CAUSE   Installation of Update 1 ZAPs ZXA038, ZXA039, ZXA040, ZXA041, ZXA042, and ZXA043 prevent you from using the SHUTDOWN command until the system is regenerated and rebooted.

SOLUTION   Regenerate and reboot the system, with the update.

2.13.18 An E$MEM Exception Occurs When Configuring A System In Background

SYMPTOM   An E$MEM exception occurs while configuring a system in the background mode.

CAUSE   Not enough memory pool is assigned.

NOTE

The default pool max for BACKGROUND is only 300K.

SOLUTION   Increase the pool max size for BACKGROUND to 384K.

2.13.19 Continuation Lines Do Not Work In CLI Commands

SYMPTOM   A E$CONTINUATION exception occurs when using continuation lines with a CLI command.

CAUSE   CLI commands do not support continuation lines.

SOLUTION   Do not use continuation lines with CLI commands.

2.13.20 DELETE Command Executed From A Submit File Acts Peculiar

SYMPTOM Executing a DELETE command from a submit file, with the DELETE command consisting of continuation lines and more than 256 characters, it appears that the first 256 characters after the DELETE command are lost.

CAUSE A problem with DELETE command.

SOLUTION Do not use the DELETE command with more than 256 characters as input. This problem will be fixed in a future update or release.

2.13.21 The File :SD:RMX286/HI/HCLUSR.P28 Is Missing

SYMPTOM The file :SD:RMX286/HI/HCLUSR.P28 is missing from the system disk.

CAUSE The file was not on the installation disk.

SOLUTION This problem is fixed in Update 2 by ZAP ZXA099.

2.13.22 Input To Terminal Port Fails To Work With 8-bit Data

SYMPTOM Data bit 7 is always cleared on input for a Human Interface user terminal which is running the Release 2 CLI.

CAUSE The Release 2 CLI sets up the serial channel associated with it to clear bit 7 on input.

SOLUTION Fixed in Update 2 by ZAPs ZXA096 and ZXA097.

2.13.23 General Protection Error Encountered While Attempting To Logon

SYMPTOM A General Protection error occurs while attempting to logon to a Dynamic Logon Terminal.

CAUSE An error in the Human Interface.

SOLUTION Will be fixed in Release 3. This problem occurs after you have added several users to the system. As a workaround, you can reduce the number of Human Interface defined users.

2.13.24 SHUTDOWN Hangs The System

SYMPTOM The system hangs if you have a maximum priority of 0 for your user and you run the SHUTDOWN command.

CAUSE An error in the SHUTDOWN command.

SOLUTION Change in user priority.

2.13.25 Problem With The Human Interface RQCGET$PARAMETER System Call

SYMPTOM The value of the "more" parameter of the RQCGET$PARAMETER system call is always set to zero.

CAUSE The predict list parameter of this system call is not in CAPS.

SOLUTION The predict list should always be in CAPS.

2.13.26 General Protection Error When A Job Is Deleted

SYMPTOM A General Protection error occurs when deleting a Human Interface job by a CONTROL-C or if in Softscope an Exit.

CAUSE The Human Interface Job's priority is higher than that of the BIOS Deletion Task.

SOLUTION Set the Connection Job Delete Priority parameter in the ICU BIOS screen to run at a higher priority than that of the Human Interface Job.

2.14 UTILITY PROBLEMS

2.14.1 LIB286 Truncates An Object File

SYMPTOM When creating a new library, LIB286 truncates an object file and issues an error message "INVALID OBJECT FILE".

CAUSE The disk has overflowed.

SOLUTION Make sure sufficient disk space is available before you invoke LIB286.

2.14.2 LIB286 Truncates Updated File - Leaves Backup File Empty

SYMPTOM When you are updating a library, LIB286 truncates the updated file and leaves the backup file empty. LIB-286 issues an error message "INVALID OBJECT FILE".

CAUSE The disk has overflowed.

SOLUTION Make sure sufficient disk space is available before you invoke LIB286.

2.14.3 Utility Generates "SYSTEM INTERFACE ERROR"

SYMPTOM The BLD286, BND286, MAP286, or OVL286 utilities generate a "SYSTEM INTERFACE ERROR". Control then transfers to the iSDM 286 Monitor, which in turn issues a stack protection error. In addition, all the other tasks running under the iRMX 286 multi-tasking system are halted.

CAUSE The disk has overflowed.

SOLUTION Make sure sufficient disk space is available before you invoke the utility.

2.14.4 PL/M-286 Or ASM286 Creates An Empty Object Module

SYMPTOM The PL/M-286 compiler or the ASM286 assembler appears to finish processing but creates an empty object module. No error messages appear at your terminal; however, the appropriate error messages do accumulate in the list or error print files.

CAUSE The disk has overflowed.

SOLUTION Make sure sufficient disk space is available before you invoke PL/M-286 or ASM286.

2.14.5 BND286 Truncates Input File To Length Zero

SYMPTOM The invocation sequence ":LANG:BND286 fname.lnk noload", will cause the input file "fname.lnk" to be truncated to length zero.

CAUSE The default output file created by BND286 consists of the input filename without the extension plus the extension ".lnk". So BND286 will create the output file "fname.lnk". The output filename is the same as the input filename.

SOLUTION Specify an output filename which is different from the input filename by the OBJECT control of BND286.

2.14.6 User-Written Exception Handler Does Not Receive Control

SYMPTOM A user-written exception handler does not receive control when an error occurs.

CAUSE The exception handler was compiled using the COMPACT compiler control.

SOLUTION Compile all exception handlers using the LARGE compiler control.

NOTE

All exception handlers should be in LARGE (FAR procedures) because they are called directly from the operating system.

2.14.7 Error When Invoking SUPER From AEDIT

SYMPTOM When you invoke the SUPER command from the AEDIT V2.1 editor, the exception 034H:E$CONN$NOT$OPEN is displayed.

CAUSE Error in the AEDIT V2.1 editor.

SOLUTION Be in SUPER before invoking the AEDIT V2.1 editor.

2.14.8 BND286 V3.1 Issues Error 140

SYMPTOM When using the NODEBUG control, BND286 V3.1 issues an error message "ERROR 140: TEXT LENGTH IS GREATER THAN SEGMENT LENGTH". This condition occurs with successive invocation of BND286 on the same file with a mix of DEBUG and NODEBUG controls.

CAUSE Error in BND286 V3.1.

SOLUTION Use the same control, either DEBUG or NODEBUG, in all the successive BND286 invocations.

2.14.9 BND286 V3.1 Gets General Protection Error

SYMPTOM BND286 V3.1 gets a General Protection error when binding an object module, whose assembly language source uses the ORG directive.

CAUSE To be determined.

SOLUTION Do not use ORG directives in modules assembled and bound under the iRMX 286 operating system.

2.14.10 Wrong Or Missing Version Numbers In iRMX 286-Based Languages/Tools

SYMPTOM When running the VERSION command on the iRMX 286-based languages and tools, a wrong version number or no version number is returned. The following list shows the values returned:

Name of Lang. or Tool	Value Displayed by VERSION
ASM286 V1.3	No version number found
BLD286 V3.1	X207
BND286 V3.1	X207
LIB286 V2.0	X207
MAP286 V2.1	X207
OVL286 V1.1	X207
ASM86 V2.0	No version number found
CREF86 V1.0	No version number found
LINK86 V2.7	No version number found
LOC86 V2.5	No version number found
OH86 V1.0	No version number found

CAUSE Error in each of the languages and tools.

SOLUTION This problem will be fixed in future releases of the languages and tools.

2.14.11 ASM86 Cannot Assemble Modules Containing Nested Macros

SYMPTOM When invoked under the iRMX 286 Operating System, ASM86 V2.0 will generate error messages on some source modules containing nested macros.

CAUSE Error in ASM86 V2.0.

SOLUTION This problem is fixed by MPL 286 in Update 0 (ZAP ZXA000).

2.14.12 EHx87.LIB Exception Handler Does Not Work Properly With iRMX

SYMPTOM The ESTATEx87 data structure gets corrupted after calling DECODE when using the NDP exception handler described in Section 5 of the *80x87 Support Library Reference Manual* within an iRMX exception handler.

CAUSE The NDP status work, obtained by executing a FNSTSW instruction or using the PM/M GET$REAL$ERROR built-in from within the exception handler does not reflect the status of the NDP when the numeric exception occurred. This is because iRMX clears the NDP error before passing control to the exception handler.

SOLUTION When iRMX passes control to an exception handler, it supplies four parameters; the exception code indicating which parameter, if any, was in error; a reserved word and a word containing the NDP status when the exception occurred. Use the NDP status word passed to the exception handler when calling DECODE.

iRMX® II OPERATING SYSTEM R3.0 **3**

This chapter lists known problems related to the iRMX II Operating System, Release 3.0. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

The structure of the chapter is as follows:

3.1   System Booting Problems

3.2   Hardware Exception Problems

3.3   Application Loader Problems

3.4   Bootstrap Loader Problems

3.5   I/O System, Device, and Driver Problems

3.6   Nucleus Problems

3.7   Universal Development Interface (UDI) Problems

3.8   Interactive Configuration Utility (ICU) Problems

3.9   Mass Storage Problems

3.10 Multiuser Problems

3.11 Debugging Problems

3.12 Pathnames Problems

3.13 Human Interface Problems

3.14 Utility Problems

3.1   SYSTEM BOOTING PROBLEMS

3.1.1   Installation System Does Not Boot

SYMPTOM   The system hangs right after the sign-on message, or never comes up, while installing the system on iAPX 286-based systems.

CAUSE   This is a problem with the iSBC™ 215G firmware Version 2.3 (Part Number 146276-001 and 146276-002).

SOLUTION   Use iSBC™ 215G firmware Version 2.4 (Part Number 146979-001 and 146979-002) or more current versions. Contact your local sales office for assistance.

CAUSE   You do not have the right stepping of the iAPX 286 processor.

SOLUTION   Use at least a B-2 stepping iAPX 286 processor. Contact your local sales office for assistance.

3.1.2   System Will Not Reboot With iSBC 215G Controller

SYMPTOM   A system that worked previously now fails to initialize when an iSBC 215G controller was added to the system.

CAUSE   The iSBC 215G controller uses a wakeup address in memory that corresponds to the wakeup port address jumpered on the iSBC 215G board. This address may be inadvertently written over by the system.

SOLUTION   Be sure your wakeup address is not included in the memory specified for the system or for the free space manager by the memory screen in the ICU. For example, the default wakeup port address for the iSBC 215G board is 100H, the memory address used is 1000H through 1100H and must be excluded from the available memory in the ICU.

3.1.3   System With Multiple iSBC 215G Boards Hangs

SYMPTOM   You have two or more iSBC™ 215G boards and when you attach a disk, the system hangs.

CAUSE   The wakeup port addresses are not a minimum of 11H apart so the 272 bytes required for each board have not been reserved.

SOLUTION   If the first board has wakeup port address of 100H the second one should have 111H and the third one should have 122H etc.

3.1.4   The System Fails To Boot When Using The 8251A Device Driver

SYMPTOM   The system does not boot when the :config:terminals file assigns a Human Interface user to a terminal attached to an iSBX 351 module.

CAUSE   To be determined.

SOLUTION   Mount the iSBX 351 board or, if the iSBX 351 module is not mounted, remove the terminal attached to the iSBX 351 module from the :config:terminals file. This problem will be fixed in a future update.

3.1.5 Using The iSBC xxxCX iLBX™ RAM Boards

SYMPTOM If your system contains more than 896 Kbytes of physical memory and you are using an iSBC xxxCX RAM Board (e.g., iSBC™ 010CX, iSBC™ 012CX, etc.), the system hangs.

CAUSE The starting address of the Human Interface user partition lies near the 896 Kbyte boundary.

SOLUTION You must configure your system so that no user partition begins within 50 Kbytes of the 896 Kbyte boundary.

3.1.6 Baud Rate Scan Of Terminal Set To 9600 Fails

SYMPTOM A baud rate scan of a terminal set to 9600 baud causes the system to fail, if your system contains either an iSBC board with clockrate greater than 6MHz (such as the iSBC 286/10A board or the iSBC 286/12 board) or an iSBC board that utilizes 1 or 0 wait-state memory and uses an 8274 Terminal Driver.

CAUSE Timing problem in interfacing with the 8274 MPC.

SOLUTION Use two or more wait-states in memory or set the terminal baud rate to something other than 9600 baud when doing the baud rate scan or configure the 8274 Terminal Driver to use a fixed baud rate.

3.1.7 "no-go" From Winchester

SYMPTOM When an iRMX system box is powered up, the SCT shows a "no-go" for the Winchester disk, but subsequent boots while system is powered up show a "go" for the disk.

CAUSE If your system has "autolock" for the disk, the SCT will fail on power up for the disk because the delay is not long enough to allow for the auto-unlock. The term "autolock" means that when the power is turned off, the heads are automatically locked, versus manual lock where a switch had to be manually set to lock the heads.

SOLUTION A future release of the SCT will have a longer delay.

3.1.8 No Response To Keyboard - No Output To Console

SYMPTOM When you attempt to boot the system, it accesses the disk, but neither responds to keyboard input nor sends any output to the console.

CAUSE The third stage Bootstrap Loader is in the same memory that is reserved for the second stage of the Bootstrap Loader.

SOLUTION If you configure a third stage Bootstrap Loader, do not locate it in the space allocated for the second stage of the Bootstrap Loader. See table.

SCT Version Number	Starting RAM Location	Size of RAM Area (in bytes)
SCT 286/300x (all versions)	24K bytes less than the largest RAM address (not to exceed 0C0000H)	21K (5400H)

CAUSE The initial task belonging to a first-level job has a missing or unexecuted END$INIT$TASK system call.

SOLUTION Check that the initial task of all first-level user jobs invokes an END$INIT$TASK system call.

CAUSE The :CONFIG:TERMINALS file has been corrupted, or has zero users allocated.

SOLUTION   Create or correct the :CONFIG:TERMINALS file.

CAUSE   The :CONFIG:USER/<user id> files are incorrect. A common error is to use both a comma (,) and a carriage return as a parameter delimiter. If a comma is followed by a carriage return, a null parameter is assigned.

SOLUTION   Correct the :CONFIG:USER/<user id> files to ensure that commas are not unintentionally followed by carriage returns.

CAUSE   Insufficient memory to complete the booting process.

SOLUTION   Check that your system has enough memory to complete booting.

CAUSE   Memory-mapped I/O addresses were not excluded from the SYS and FSM memory.

SOLUTION   Exclude memory-mapped I/O addresses from iRMX II memory using the MEMORY FOR SYSTEM screen of the ICU.

CAUSE   The system has been configured to use the System Debug Monitor (iSDM™), but the iSDM monitor has not been physically installed in the system or it has been placed at an invalid address.

SOLUTION   Install the iSDM monitor. For information about how to install and/or use the iSDM monitor, refer to the *iSDM™ System Debug Monitor Installation and Configuration Guide.*

CAUSE   The system device name in the Bootstrap Loader does not have a corresponding device name in a Device Unit Information entry within the system definition file.

SOLUTION   Using the ICU, make sure that the device name of the booting device exists in a Device Unit Information entry. For example, "W" for the Winchester disk.

CAUSE   There is a conflict over an interrupt. For example, two or more interrupts are assigned to the same level.

SOLUTION   Check the board jumpers and the configuration definition file for two interrupts assigned to the same level.

CAUSE   The interrupt from a device is not getting to the 8259A Programmable Interrupt Controller.

NOTE

The Bootstrap Loader does not use interrupts.

SOLUTION   Check that the interrupt is actually getting to the 8259A (PIC). Refer to the hardware reference manual for your Single Board Computer (iSBC).

CAUSE   The disk has been formatted using the iRMX 86 Release 7 FORMAT and the second stage of the bootstrap loader for the iRMX 86 Release 7 system is not compatible with the iRMX II system.

SOLUTION   Format the disk using iRMX II FORMAT.

3.1.9 ROM System Does Not Come Up

SYMPTOM   When a ROM based iRMX II operating system is powered up, it does not come up.

CAUSE   The iSBC 286/10A or iSBC 286/10 board is not jumpered correctly.

SOLUTION   To specify four 27512 EPROM devices, set up jumpers 62 through 91 as specified in the 27512 (64K x 8 EPROM) example in Appendix E of the *iSBC 286/10A or iSBC 286/10 Hardware Reference Manual.*

To specify a starting memory address and memory size for local memory, use primary decode option 3. The jumpers required are:

Default Configuration	Jumpers for Primary Decode Option 3
E218 - E219 installed	E218 - E219 installed
E220 - E221 removed	E220 - E221 installed

To specify memory size/justification for local memory, use secondary option 3. The jumpers required are:

Default Configuration	Jumpers for Primary Decode Option 3
E51 - E59 removed	E51 - E59 removed
E50 - E58 removed	E50 - E58 installed
E49 - E57 installed	E49 - E57 installed

3.1.10 EIOS Initialization Error 40h Encountered

SYMPTOM EIOS Initialization Error 40h is encountered when bootstrap loading an iRMX II operating system.

CAUSE The cause is under investigation.

SOLUTION Re-boot your iRMX II operating system. Will be fixed in a future release.

3.1.11 Error 118 Received From BLD286 Utility

SYMPTOM The BLD286 utility generates the following error message during system configuration.

```
*** ERROR 118: Input Segments Exceed Target Memory Processing Aborted
```

CAUSE The amount of memory needed for the iRMX 286 operating system code and data segments has exceeded the amount of memory given to it during system configuration.

SOLUTION Use the ICU286 utility and increase the system memory on the memory for system screen. Decrease the memory specified on the memory for free space manager screen by the same amount. Refer to the segment description portion of the BLD286 map file (<boot_file>.mp2).

3.1.12 Checksum Error Encountered While Bootloading The Operating System

SYMPTOM A checksum error is encountered when you bootload the iRMX II operating system.

CAUSE A file was specified in the first stage of the bootstrap loader that was not the third stage of the bootstrap loader.

SOLUTION Always specify the third stage of the bootstrap loader when bootloading your system.

3.1.13 Problems With Installation Of Update 2

SYMPTOM Many error messages may be generated during the installation process of Update 2.

CAUSE The LIB86 utility is required for the installation of Update 2.

SOLUTION Make sure you have the LIB86 utility before installing Update 2.

3.1.14 Memory Warning Received During the Logon Process

SYMPTOM The operating system may display a WARNING message during the logon process, informing the user that it cannot assign it the minimum memory it has requested.

CAUSE The maximum memory partition-size value of the user attribute file was smaller than the minimum memory partition-size.

SOLUTION Contents of the user attribute file before and after the problem:

Before: 2000,512,141,:sd:user/super After : 512,2000,141,:sd:user/super

3.1.15 An "INTERRUPT 3" May Be Encountered When Bootloading The Operating System

SYMPTOM An "INTERRUPT 3" may be encountered when bootloading the iRMX II Operating System from a single high density, 96 TPI boot diskette.

CAUSE Problem in the iRMX II bootstrap loader. As a workaround you must use the generic third stage of the bootstrap loader for your specific board.

SOLUTION This problem will be fixed in a future update/release of the iRMX II product.

3.2 HARDWARE EXCEPTION PROBLEMS

3.2.1 General Protection In A System Call

SYMPTOM You get a general protection error within a system call, even though the parameters passed to the call are correct.

CAUSE The application program is bound to the wrong interface library. For example, a LARGE program is bound to a COMPACT interface library.

SOLUTION Check your program size control statement and the bind command to make sure they are compatible.

3.2.2 General Protection In An Application Program

SYMPTOM When running an application program, you get a general protection error.

CAUSE You have an invalid data segment or you have crossed the boundary of the segment.

SOLUTION Make the necessary correction to your program.

NOTE

The iSDM command "ddt(ds)" will display the data segment characteristics. By examining the data segment you can identify the problem.

CAUSE You have an invalid extra segment or you have crossed the boundary of the segment.

SOLUTION Make the necessary correction to your program.

NOTE

The iSDM command "ddt(es)" will display the extra segment characteristics. By examining the extra segment you can identify the problem.

3.2.3 Stack Protection In An Application Program

SYMPTOM   When running an application program you get a stack protection error.

CAUSE   You have an invalid stack segment or the stack has overflowed.

SOLUTION   Make the necessary correction to your program.

NOTE

The iSDM command "ddt(ss)" will display the stack segment characteristics. By examining the stack segment you can identify the problem.

CAUSE   The PL/M-286 compiler did not create a stack and the application loader did not create a stack big enough for the execution of the program.

SOLUTION   Use debug control when compiling.

3.2.4 Stack Protection Error When Running BLD286

SYMPTOM   "Stack Protection Error When Running BLD286" When running BLD286 on a system that does not have enough disk space, a stack protection exception is encountered. The builder displays the following messages a number of times before it gets a protection error:

```
INTERFACE ERROR
0029 : E$SPACE
FILE : ROM.MP2
PROCESS ABORTED
```

CAUSE   The disk does not have enough space to contain the output files from the build operation.

SOLUTION   Make sure your disk has over 1 Mbyte free space before starting the ICU. This problem will be fixed in a future release of BLD286.

3.2.5 Third Stage Gets DIVIDE-BY-ZERO Error

SYMPTOM   The third stage of the Bootstrap Loader gets a DIVIDE-BY-ZERO error.

CAUSE   Your system was not reset before you booted your iRMX II operating system.

SOLUTION   Reset the system by pressing the reset button on your system's front panel or by whatever means you have built into your hardware.

CAUSE   You have configured the IDT with too few slots.

SOLUTION   Check your definition file and correct the number of IDT entries in the Nucleus screen of the ICU.

CAUSE   You are using an iRMX 86 operating system Bootstrap Loader second stage. In other words, you are booting from a disk which was formatted using the iRMX 86 FORMAT command.

SOLUTION   Reformat the disk using iRMX II FORMAT command with BOOTSTRAP option only the second stage of the Bootstrap Loader is written by this option; the other files on the disk are uneffected.

3.3   APPLICATION LOADER PROBLEMS

3.3.1   Wrong Message Returned To MSG$MBOX In RQ$S$LOAD$IO$JOB

SYMPTOM   When using the RQSLOADIOJOB system call, the mailbox specified in the MSG$MBOX parameter receives the wrong message.

CAUSE   The mailbox specified in the MSG$MBOX parameter of the RQ$S$LOAD$IO$JOB call is used internally by the loader to receive the loader result segment from RQALOADIOJOB. This causes several problems:

1. The user cannot specify selector$of(NIL) for the MSG$MBOX parameter as in RQ$CREATE$IO$JOB.

2. The user can receive the wrong message. If the user has a separate task to receive exit messages from I/O jobs and this task does a RQ$RECEIVE$MESSAGE at the MSG$MBOX before another task calls RQSLOADIOJOB, the user will receive the loader result segment and the system will crash.

3. The user can receive the wrong message. If the user has 2 RQSLOADIOJOB calls that specify the same MSG$MBOX, the loader can receive the exit message from the first I/O job while expecting to get the loader result segment from the second load. This prevents delivery of the exit message and causes the second load to terminate without success.

SOLUTION   As a workaround, avoid the following:

1. Specify NIL for the MSG$MBOX of the RQ$S$LOAD$IO$JOB

2. Leave a task waiting for a message at a mailbox and then use the same mailbox in a RQSLOADIOJOB call.

This problem will be fixed in a future update or release.

3.3.2   General Protection Error When Running Program

SYMPTOM   When trying to run a large program you encounter a general protection error.

CAUSE   An error in the application loader regarding the processing of fix-up records.

SOLUTION   Apply II.3, Class B ZAPS #078, 097, 100, 101. They provide a complete fix.

3.4   BOOTSTRAP LOADER PROBLEMS

3.4.1   Third Stage Does Not Issue The First Message

SYMPTOM   While bootloading, the message, NOW IN PROTECTED MODE, is not issued and the system hangs.

CAUSE   The load file you specified with a .286 extension or the default load file you used, /SYSTEM/RMX86.286 (if no name is specified) was not found on the volume.

SOLUTION   Check the volume to make sure the file specified or the default file /SYSTEM/ RMX86.286 exists.

CAUSE   The pathname for the third stage has a period in it other than the period for the file name extension.

SOLUTION   Rename the directory or the filename in the pathname of the third stage, so that the

name does not contain a period, except the period for the extension of the third stage's file name.

For example, if the third stage has a pathname /BOOT/MY.DIR/SYS286.3RD, change it to /BOOT/MYDIR/SYS286.3RD.

3.4.2 Third Stage Does Not Issue The Second Message

SYMPTOM While bootloading, the message, TARGET FILE LOADED, is not issued and the system hangs.

CAUSE The file containing the system was incorrect.

SOLUTION Check the file and regenerate the system if necessary.

CAUSE The third stage was loading the load file on top of it.

SOLUTION Use the memory screen of the ICU to correct your memory allocation. By default, the second stage is located at B8000H, BBFFFH - C0000H, the device generic and specific third stages are located at BC000H - C0000H. Your system memory should not include these locations. However, the free space manager can include this memory.

3.4.3 Third Stage Gets A DIVIDE-BY-ZERO Error

SYMPTOM The third stage of the Bootstrap Loader gets a DIVIDE-BY-ZERO error.

CAUSE Your system was not reset before you booted your iRMX 286 operating system.

SOLUTION Reset the system by pressing the reset button on your system's front panel or by whatever means you have built into your hardware.

CAUSE You have configured the IDT with too few slots.

SOLUTION Check your definition file and correct the number of IDT entries in the Nucleus screen of the ICU.

CAUSE You are using an iRMX 86 operating system Bootstrap Loader second stage. In other words, you are booting from a disk which was formatted using the iRMX 86 FORMAT command.

SOLUTION Reformat the disk using the iRMX 286 FORMAT command with the BOOTSTRAP option.

3.4.4 Checksum Error Encountered While Bootloading The Operating System

SYMPTOM A checksum error is encountered when you bootload the iRMX II operating system.

CAUSE A file was specified in the first stage of the bootstrap loader that was not the third stage of the bootstrap loader.

SOLUTION Always specify the third stage of the bootstrap loader when bootloading your system.

3.4.5 You Cannot Bootload The iRMX Operating System From A Diskette

SYMPTOM You cannot bootload the iRMX II.3 Operating System from a diskette that has been fomatted with more than 8 sectors per track.

CAUSE A problem with the Bootstrap Loader.

SOLUTION This problem was fixed in Update 2 by class A ZAPs ZXA049, ZXA0183, ZXA0184, ZXA0185, and ZXA0186.

3.4.6 An "INTERRUPT 3" May Be Encountered When Bootloading The Operating System

SYMPTOM An "INTERRUPT 3" may be encountered when bootloading the iRMX II operating system from a single high density, 96 TPI boot diskette.

CAUSE Problem in the iRMX II bootstrap loader. As a workaround you must use the generic third stage of the bootstrap loader for your specific board.

SOLUTION This problem will be fixed in a future update/release of the iRMX II product.

3.5 I/O SYSTEM, DEVICE, AND DRIVER PROBLEMS

3.5.1 System Time Not Current When CONTROL-C Out Of TIME Command

SYMPTOM If you abort the default r?logon program at signon time via a CONTROL-C, the default Date/Time of the system does not automatically revert to the last time the directory :SYSTEM: was accessed.

CAUSE The system gets its time from the :SYSTEM: directory if the operator does not enter time during signon time. To improve performance, iRMX II does not update a file's access parameter unless the file was modified. Because of this, the access time of :SYSTEM: directory reflects the last time it was modified.

SOLUTION Enter time each time you logon. Run "SHUTDOWN" before you turn power off or reset your system.

3.5.2 iSBC 188/48 Board Ignores Characters With Parity Errors

SYMPTOM The iSBC 188/48 driver does not pass thru characters with input parity errors.

CAUSE A feature of the iSBC 188/48 firmware.

SOLUTION Use a different device. For example, an iSBC 544A board may be used.

3.5.3 iSBC 188/48 and iSBC 188/56 Board Problem

SYMPTOM A General Protection error is encountered in a system that uses the iSBC 188/48 or iSBC 188/56 board.

CAUSE A problem with the iSBC 188/48 and iSBC 188/56 boards (if IC U-69 is not labeled 451012).

SOLUTION Return the board to the repair center for upgrade.

3.5.4 Line Printer Hangs System

SYMPTOM The line printer hangs the system when the printer is OFFLINE and then turned back ONLINE.

CAUSE There is a problem with ZAP's ZAXAB2 and ZAXAB3.

SOLUTION Enter a CONTROL-C to stop the line printer from hanging the port. Do not apply the ZAPs or make sure your printer is ONLINE when you copy a file to it. This problem will be fixed in a future update or release.

3.5.5 Translation Of Escape Sequence #18 Does Not Work

SYMPTOM Translation of escape sequence 18 does not work.

CAUSE A problem in the Terminal Support Code.

SOLUTION This problem will be fixed in a future update or release.

3.5.6 Cannot Use iSBX 251 At Addresses Other Than 80H

SYMPTOM The iSBX 251 driver will not work if you move the iSBX 251 driver data port address, anywhere other than a data port address of 80H.

CAUSE A problem with the ICU where the IDEVC.A28 file does not reflect the change of data port address made when configuring the driver.

SOLUTION Edit the IDEVC.A28 file to reflect the correct data port address. This problem will be fixed in a future update or release.

3.5.7 RQSCREATE$FILE System Call May Cause A General Protection Error

SYMPTOM Using the RQSCREATE$FILE system call may result in a General Protection error.

CAUSE The path pointer parameter for this system call was set to NIL. A problem with the RQSCREATE$FILE system call.

SOLUTION This problem was fixed in Update 1 by class A ZAP ZXA040. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.5.8 RQSRENAME$FILE System Call May Cause A General Protection Error

SYMPTOM Using the RQSRENAME$FILE system call may result in a General Protection error.

CAUSE The path pointer parameter for this parameter was set to NIL. A problem with the RQSRENAME$FILE system call.

SOLUTION This problem was fixed in Update 1 by class A ZAP ZXA041. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.5.9 RQSCREATE$DIRECTORY System Call May Cause A General Protection Error

SYMPTOM   Using the RQSCREATE$DIRECTORY system call may result in a General Protection error.

CAUSE   The path pointer parameter for this system call was set to NIL. A problem with the RQSCREATE$DIRECTORY system call.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA039. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.5.10 RQSDELETE$FILE System Call May Cause A General Protection Error

SYMPTOM   Using the RQSDELETE$FILE system call may result in a General Protection error.

CAUSE   The path pointer parameter for this system call was set to NIL. A problem with the RQSDELETE$FILE system call.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA038. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.5.11 RQSGET$FILE$STATUS System Call May Cause A General Protection Error

SYMPTOM   Using the RQSGET$FILE$STATUS system call may result in a General Protection error.

CAUSE   The path pointer parameter for this system call was set to NIL. A problem with the RQSGET$FILE$STATUS system call.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA037. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.5.12 Occasional Disk File Corruption

SYMPTOM   Occasional disk file corruption occurs when the number of blocks in a file exceeds 64K.

CAUSE   Problem with the named file driver of the Basic I/O System.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA029. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.5.13 Unresolved External Errors When Configuring Terminal Drivers

SYMPTOM   User-configured terminal drivers cannot be added to the system because of unresolved external errors during system configuration.

CAUSE   Problem with the Interactive Configuration Utility Templete File.

SOLUTION   This problem was fixed in Update 1 by class A ZAPs ZXA015, ZXA016, and ZXA017. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.5.14 The Terminal May Hang While Scrolling Through A File Using AEDIT

SYMPTOM The terminal may hang occassionally, while scrolling through a file using AEDIT.

CAUSE Error in the Terminal Communication Controller (TCC) driver.

SOLUTION This problem was fixed in Update 2 by class A ZAP ZXA030.

3.5.15 The 8274 Terminal Driver Is Too Slow When Handling Interrupts

SYMPTOM The 8274 terminal driver consumed over 500 microseconds of CPU time for each interrupt.

CAUSE Problem with the 8274 terminal driver.

SOLUTION This problem was fixed in Update 2 by class A ZAP ZXA043.

3.5.16 A General Protection Exception Is Encountered When Using RAASPECIAL System Call

SYMPTOM A general protection exception is encountered when setting terminal attributes using the RQASPECIAL system call.

CAUSE Error in the the iRMX II.3 BIOS.

SOLUTION This problem was fixed in Update 2 by class A ZAP ZXA044.

3.5.17 A General Protection Exception Is Encountered When Using RQSSPECIAL System Call

SYMPTOM A general protection exception is encountered when setting terminal attributes using the RQSSPECIAL system call.

CAUSE Problem with the iRMX II.3 BIOS.

SOLUTION This problem was fixed in Update 2 by class A ZAP ZXA044.

3.5.18 The System May Hang When Bootloading From A SCSI Device

SYMPTOM The system may hang during the bootload process if a SCSI device is used.

CAUSE A problem with the Bootstrap Loader.

SOLUTION This problem was fixed in Update 2 by class A ZAPs ZXA0181 and ZXA0182.

3.5.19 You Cannot Bootload The iRMX Operating System From A Diskette

SYMPTOM You cannot bootload the iRMX II.3 Operating System from a diskette that has been fomatted with more than 8 sectors per track.

CAUSE A problem with the Bootstrap Loader.

SOLUTION This problem was fixed in Update 2 by class A ZAPs ZXA049, ZXA0183, ZXA0184, ZXA0185, and ZXA0186.

3.5.20 An E$IO_HARD Is Encountered While Formatting A Diskette

SYMPTOM An E$IO_HARD is encountered while formatting a floppy diskette.

CAUSE During system configuration, the NS (number of sectors/track) parameter was set to 9 or more. Problem with the Bootstrap loader.

SOLUTION This problem was fixed in Update 2 by class A ZAPs ZXA049, ZXA0183, ZXA0184, ZXA0185, and ZXA0186.

3.5.21 CONTROL-S Can Hang A Terminal

SYMPTOM While scrolling through a file using AEDIT, if you enter a CONTROL-S you may not be able to resume using a CONTROL-Q.

CAUSE Problem under investigation.

SOLUTION This problem will be fixed in a future update or release.

3.5.22 A General Protection Exception Is Encountered When Deleting A Task

SYMPTOM A General Protection Exception may be encountered when trying to delete a task.

CAUSE Problem in the Nucleus.

SOLUTION This problem was fixed in Update 2 by class A ZAP ZXA054.

3.5.23 Your Response Time To HI Commands Seems Too Long

SYMPTOM You response time to HI commands seem too long.

CAUSE The default priority of the SUPER user is 141. Users, other than SUPER, are created with a priority of 142. If SUPER user is active in a multi-user system, other users may notice a degradation in response time.

SOLUTION Edit the :CONFIG:USER/SUPER file and change the priority to 142. Logout and then login again.

3.5.24 Problems Using The iSBC 221 Peripheral Disk Controller

SYMPTOM Various problems may be encountered when configuring and using the iSBC 221 Peripheral Disk Controller.

CAUSE Problem with the Mass Storage Controller (MSC) driver.

SOLUTION This problem was fixed in Update 2 by class A ZAPs ZXA049, ZXA0183, ZXA0184, ZXA0185, and ZXA0186. Refer to the iRMX II.3 Update READ.ME file for complete information about the iSBC 221 Peripheral Disk Controller.

3.5.25 General Protection Exception Encountered When Saving A File Using AEDIT

SYMPTOM "General Protection Exception encountered when saving a file using AEDIT" A General Protection exception may be encountered when saving a file on a floppy using AEDIT.

CAUSE The floppy was removed from the diskette drive, before the save command was issued. The problem is under investigation.

SOLUTION Ensure that a floppy diskette is in the drive and the floppy drive door is closed.

3.5.26 Line Printer Very Slow

SYMPTOM The line printer prints a single line every five seconds.

CAUSE A problem in the iRMX II.3 nucleus and device driver.

SOLUTION This problem was fixed by Class B zaps ZAP080, ZAP082, and ZAP083. U.S. customers must contact TIPS. European customers must contact their local sales office.

3.5.27 Problem With The iSBC 186/410 Device In Full Duplex Mode

SYMPTOM Input is extremely slow compared to output on the iSBC 186/410 device when used in full duplex mode.

CAUSE Problem with the iSBC 186/410 device driver.

SOLUTION This problem will be fixed in a future update/release of the iRMX II product.

3.5.28 Cannot Use The iRMX Virtual Terminal

SYMPTOM If a user invokes VT and logs on to a remote iRMX II.3 system, does some work, logs out, and then terminates the VT session (via "~."), he is unable to VT to that system again.

CAUSE Problem with Update 2 of the iRMX II.3 Operating System.

SOLUTION As a workaround, ensure that you terminate the Virtual Terminal session (via "~.") before you log off the system. This problem has been fixed by Class B zap ZAP076. U.S. customers may contact TIPS. European customers may contact their local sales office.

3.5.29 An E$IOMEM Exception May Be Encountered When Doing An "ATTACHDEVICE"

SYMPTOM An E$IOMEM exception may be encountered when doing an "ATTACHDEVICE".

CAUSE The user has not received the requested amount of memory during the "LOGON" process and therefore has insufficient memory.

SOLUTION Ensure that the "minimum partition-size" parameter of the user attribute file (:config:user/<user-name>) is at least 512 and this value is less than the "maximum partition-size" parameter value.

3.5.30 Cannot Input More Than 255 Bytes At A Time When Using The Computer-To-Computer Feature

SYMPTOM One cannot input more than 255 bytes at a time when using the computer-to-computer feature of the iRMX II operating system.

CAUSE This is a limitation of the iRMX II operating system.

SOLUTION Repeated input statements will allow the program to obtain the data being sent to it.

3.6 NUCLEUS PROBLEMS

3.6.1 System Is In Infinite Loop

SYMPTOM   If the SDB is not the exception handler in your system and the exception mode is "never", a DIVIDE-BY-ZERO error in your application program will cause an infinite loop to be executed.

CAUSE   The system default exception handler and the system alternative exception handler do not increment the IP register past the offending instruction.

SOLUTION   Use the SDB as the exception handler during the development phase and write a user exception handler if DIVIDE-BY-ZERO can occur in the application.

3.6.2 Cannot Set Exception Handlers For Hardware Exceptions

SYMPTOM   RQSETEXCEPTION$HANDLER will not install a user-written exception handler for a hardware exception.

CAUSE   The iRMX II Release 3.0 system does not provide this capability.

SOLUTION   This problem will be fixed in a future release.

3.6.3 RQ$RESUME$TASK Call Gets E$STATE

SYMPTOM   A task which makes a RQ$DISABLE$DELETION call to protect itself from deletion and then makes a RQ$SUSPEND$TASK call to suspend itself, cannot be resumed by a RQ$RESUME$TASK call. The RQ$RESUME$TASK returns a condition code of E$STATE which indicates that the task was not suspended.

CAUSE   Error in NUC.LIB.

SOLUTION   This problem will be fixed in a future update.

3.6.4 Exceptions Are Not Caught By The Exception Handler

SYMPTOM   Environmental exceptions are not caught by the exception handler.

CAUSE   The UDI exception handler acts only on programmer exceptions, and when binding your program you specified the UDI Interface Library before the iRMX 286 Interface Library.

SOLUTION   In your BND286 statement specify the iRMX 286 Interface Library before the UDI Interface Library.

3.6.5 E$STATE Exception Never Returned By RQ$RESUME$TASK

SYMPTOM   An E$STATE exception is not returned if you resume a non-suspended task with the Nucleus call RQ$RESUME$TASK.

CAUSE   An error in the Nucleus.

SOLUTION   Will be fixed in a future release.

3.6.6 Problem With The RQ$RESUME$TASK() System Call

SYMPTOM   If you resume a non-suspended sleeping task with the nucleus system call RQ$RESUME$TASK, the nucleus does not return an E$STATE exception.

CAUSE   Problem with the RQ$RESUME$TASK system call.

SOLUTION   This problem will be fixed in a future update or release.

3.6.7 RQEOBJECT$ACCESS Returns An E$BAD$ADDRESS Exception

SYMPTOM   The RQEOBJECT$ACCESS nucleus system call returns a EBADADDRESS exception if the ACCESS$PTR parameter of this call points to a structure that is located at the end of the data segment.

CAUSE   Problem with the RQEOBJECT$ACCESS system call.

SOLUTION   This problem will be fixed in a future update or release.

3.6.8 RQGETINTERCONNECT System Call May Cause A General Protection Error

SYMPTOM   You encounter a General Protection error when using the RQGETINTERCONNECT system call.

CAUSE   An ENOTCONFIGURED exception should have been returned. A problem with the Interactive Configuration Utility Template File.

SOLUTION   This problem was fixed in Update 1 by class A ZAPs ZXA015, ZXA016, and ZXA017. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.6.9 RQSETINTERCONNECT System Call May Cause A General Protection Error

SYMPTOM   You encounter a General Protection error when using the RQSETINTERCONNECT system call.

CAUSE   An ENOTCONFIGURED exception should have been returned. A problem with the Interactive Configuration Utility Template File.

SOLUTION   This problem was fixed in Update 1 by class A ZAPs ZXA015, ZXA016, and ZXA017. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.6.10 E$CONTEXT Received From Buffer Pool System Calls

SYMPTOM   An E$CONTEXT is received when executing buffer pool system calls.

CAUSE   The buffer pools and the buffers that are used with the buffer pool system calls were not created by the job they are used in.

SOLUTION   Make sure that the buffer pools and the segments (buffers) that are used with the buffer pool system calls are created by the job they are used in.

3.6.11 The iRMX Operating System Does Not Support The 80287 Emulator

SYMPTOM   The iRMX Operating System does not support the 80287 Emulator. Incorrect Floating Point values may be returned.

CAUSE   The 80287 Emulator has non-reentrant routines. Therefore the state of the Emulator will not be saved during a task switch.

SOLUTION   You must either obtain and configure an 80287 component or synchronize your tasks (i.e, using SEMAPHORES) in a way that only one task uses the Emulator at any time.

3.6.12 RQ$DELETE$BUFFER$POOL System Call Always Returns An E$OK

SYMPTOM   A call to the RQ$DELETE$BUFFER$POOL Nucleus system call to delete a buffer pool that is attached to a port always returns a condition code of E$OK.

CAUSE   Problem in the iRMX Nucleus.

SOLUTION   This problem will be fixed in a future update or release.

3.6.13 Problem With The RQ$SEND$RSVP Nucleus System Call

SYMPTOM   Control does not return from the RQ$SEND$RSVP nucleus system call.

CAUSE   Existing, but incorrect "port$tkn" and "socket" parameters were used for this system call.

SOLUTION   Use valid and correct "port$tkn" and "socket" parameters.

3.6.14 General Protection Exception Encountered When Using The RQ$DELETE$DESCRIPTOR Nucleus System Call

SYMPTOM   A General Protection Exception may be encountered when the RQ$DELETE$DESCRIPTOR Nucleus system call is used.

CAUSE   A Problem in the iRMX II Nucleus.

SOLUTION   This problem will be fixed in a future update/release of the iRMX II product.

3.6.15 An E$INTERRUPT$OVERFLOW Message Encountered

SYMPTOM   An E$INTERRUPT$OVERFLOW message may be encountered from the Nucleus RQ$TIMED$INTERRUPT system call.

CAUSE   Problem with the iRMX II.3 Nucleus.

SOLUTION   This problem will be fixed in a future update/release of the iRMX II product.

3.6.16 NUCLEUS INITIALIZATION ERROR: 0002H Encountered

SYMPTOM   A "NUCLEUS INITIALIZATION ERROR : 0002H" may be encountered when bootloading the iRMX II operating system.

CAUSE   All available memory was designated as system memory. Therefore, the operating system had insufficient memory for its initialization.

SOLUTION   Use the interactive configuration utility, and change the system memory screen to allow sufficient memory for the system to come up and initialize.

3.6.17 A General Protection Exception May Be Encountered When Executing A RQ$RECEIVE$DATA Nucleus System Call

SYMPTOM A general protection exception may be encountered when executing an RQ$RECEIVE$DATA nucleus system call.

CAUSE Problem in iRMX II nucleus. As a workaround you must ensure that the receive buffer size is at least 128 bytes in length.

SOLUTION This problem will be fixed in a future update/release of the iRMX II product.

3.7 UNIVERSAL DEVELOPMENT INTERFACE (UDI) PROBLEMS

3.7.1 CONTROL-C Handler Gets Stack Protection Error

SYMPTOM A user-written CONTROL-C handler gets a Stack Protection error.

CAUSE The CONTROL-C handler was compiled using the COMPACT model of compilation.

SOLUTION Compile the CONTROL-C handler using the LARGE model of compilation.

3.7.2 Run Time Error 0001H Encountered In An I/O Job

SYMPTOM When making a FORTRAN written application an I/O job, you receive a run time exception of 0001H.

CAUSE A problem with the UDI.

SOLUTION Lookup "R?HCNAME" in the CLI's job and catalog it in the I/O job's directory under the same name. This problem will be fixed in a future update or release.

3.8 INTERACTIVE CONFIGURATION UTILITY (ICU) PROBLEMS

3.8.1 BLD286 Issues "REFERENCE TO EMPTY SEGMENT" Error Message

SYMPTOM If a user module contains only large model reentrant procedures and this module is specified as a user module in the ICU, then BLD286 issues an error message "REFERENCE TO EMPTY SEGMENT".

CAUSE Reentrant procedures use the stack for their data. As a result, the data segment remains empty. In the large model, this empty data segment does not get combined with the data segments of other modules.

SOLUTION Declare a "dummy" variable in the module at the module level, so that the data segment is not empty.

3.8.2 Changes To Definition File Were Not Saved

SYMPTOM After generating a system, changes made in the ICU session did not get changed in the definition file.

CAUSE The ICU tried to access files without having the correct access rights for those files.

SOLUTION Refer to chapter two of the *Extended iRMX II.3 Configuration Guide* for correct procedures on access rights of ICU files.

3.8.3 Problem With The UDS Utility

SYMPTOM   You encounter bad data in the .tpl input file when using the UDS utility.

CAUSE   Problem with the UDS utility.

SOLUTION This problem was fixed in Update 1 by class A ZAP ZXA035. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.8.4 Problem With The 86-based ICU286 Utility

SYMPTOM   The 86-based ICU286 utility fails.

CAUSE   Problem with the 86-based ICU286 utility.

SOLUTION This problem was fixed in Update 1 by class A ZAP ZXA034. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.8.5 Unresolved External Errors When Configuring Terminal Drivers

SYMPTOM   User-configured terminal drivers cannot be added to the system because of unresolved external errors during system configuration.

CAUSE   Problem with the Interactive Configuration Utility Template File.

SOLUTION This problem was fixed in Update 1 by class A ZAPs ZXA015, ZXA016, and ZXA017. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.8.6 RQGETINTERCONNECT System Call May Cause A General Protection Error

SYMPTOM   You encounter a General Protection error when using the RQGETINTERCONNECT system call.

CAUSE   An ENOTCONFIGURED exception should have been returned. A problem with the Interactive Configuration Utility Template File.

SOLUTION This problem was fixed in Update 1 by class A ZAPs ZXA015, ZXA016, and ZXA017. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.8.7 RQSETINTERCONNECT System Call May Cause A General Protection Error

SYMPTOM   You encounter a General Protection error when using the RQSETINTERCONNECT system call.

CAUSE   An ENOTCONFIGURED exception should have been returned. A problem with the Interactive Configuration Utility Template File.

SOLUTION This problem was fixed in Update 1 by class A ZAPs ZXA015, ZXA016, and ZXA017. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.8.8 Assembly Error During System Configuration

SYMPTOM An assembly error is encountered during the configuration process if NO Intel devices are included in a system containing the BIOS.

CAUSE Problem with the Interactive Configuration Utility Template File.

SOLUTION This problem was fixed in Update 1 by class A ZAPs ZXA015, ZXA016, and ZXA017. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.8.9 Receiving NDP Exceptions When Running Programs Under Human Interface

SYMPTOM You may encounter NDP exceptions when you run a program from the Human interface.

CAUSE The NPX parameter in the HI JOBS screen is set to NO.

SOLUTION The standard ICU definition files 286100A.DEF and 386100.DEF have the NPX parameter set to YES on the hardware screen, but set to NO on the HI JOBS screen. Therefore, if you have commands that will be using floating point instructions and will be run from the Human Interface you must use the ICU286 utility to set the NPX parameter in the HI JOBS screen to YES.

3.8.10 Problems Encountered When Using The MKSYS Submit File

SYMPTOM The MKSYS submit fails during system generation using PL/M-286 V3.1 and/or ASM286 V1.3.

CAUSE Not enough memory available for PLM286 V3.1 when run from the MKSYS submit file.

SOLUTION Increase your minimum user memory partition size to 420K bytes.

3.8.11 Cannot Have More Than 100 Operating System Extensions

SYMPTOM The ICU286 utility will not allow you to enter more than 100 Operating System extensions.

CAUSE Problem with the ICU286 utility.

SOLUTION This problem was fixed by Class B ZAP072. U.S. customers may contact TIPS. European customers customers may contact their local sales office.

3.9 MASS STORAGE PROBLEMS

3.9.1 Cannot Format iSBX 251 Multimodule

SYMPTOM The user cannot format an iSBX 251 Magnetic Bubble Memory Multimodule.

CAUSE Improper jumpering on iSBX 251 multimodule.

SOLUTION On the iSBX 251 multimodule, remove jumper W5 and install jumper W6. For more information on jumpering the iSBX 251 multimodule, refer to the *iSBX™ 251 Multimodule Technical Reference Manual*.

3.9.2 Data Loss On iSBX 251 Driver

SYMPTOM   Data loss occurs on an iSBX 251 driver.

CAUSE   The CPU must service all interrupts from the iSBX 251 driver within 1.5 milliseconds. Failure to respond within this time frame could cause the iSBX 251 driver to lose data.

SOLUTION   Solutions are as follows:

1. Do not mount the iSBX 251 Multimodule on a board with a clockrate greater than 6MHz (such as an iSBC 286/12).

2. Do not mount the iSBX 251 Multimodule on a CPU board that uses less than two wait-states on the iLBX.

3. Do not use a terminal that uses the terminal support code.

3.9.3 Data Loss On Mass Storage Device

SYMPTOM   Data loss occurs on a mass storage device.

CAUSE   System data buffers have not been written to the storage device prior to power loss or pressing the reset button.

SOLUTION   Following disk access, wait for the update timeout (usually two seconds) before pressing the reset button or powering down the system.

NOTE

It is good practice to close all files and use the system call A$UPDATE while accessing disk files under program control. When rebooting or powering down your system, always run shutdown first.

3.9.4 Data Loss On A Diskette

SYMPTOM   Data loss occurs on a diskette.

CAUSE   Some diskette drives do not have a Ready Line to inform the operating system when diskettes are changed; this applies to 5.25" drives, slim-line 8" drives, and micro floppy drives.

SOLUTION   Use the ATTACHDEVICE command after a new diskette has been inserted and use the DETACHDEVICE command before removing a diskette from the drive.

3.9.5 Cannot Use iSBX 251 At Addresses Other Than 80H

SYMPTOM   The iSBX 251 driver will not work if you move the iSBX 251 driver data port address, anywhere other than a data port address of 80H.

CAUSE   A problem with the ICU where the IDEVC.A28 file does not reflect the change of data port address made when configuring the driver.

SOLUTION   Edit the IDEVC.A28 file to reflect the correct data port address. This problem will be fixed in a future update or release.

3.9.6 Occasional Disk File Corruption

SYMPTOM   Occasional disk file corruption occurs when the number of blocks in a file exceeds 64K.

CAUSE   Problem with the named file driver of the Basic I/O System.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA029. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.9.7 A "VOLUME NOT MOUNTED" Message Encountered While Installing The iRMX II.3 Operating System

SYMPTOM   A "VOLUME NOT MOUNTED" message may be encountered while installing the iRMX II.3 Operating System.

CAUSE   This message is generated when you try to format your hard disk using the "form_disk" submit file. The command fails when the "changesd" command is executed to switch the system device from the floppy to the hard disk. This problem was caused by an incorrect jumpering of the disk drive.

SOLUTION   Make sure your disk drive is jumpered correctly. For disk drive jumper information, refer to the hardware reference manual for your particular brand of drive.

3.10 MULTIUSER PROBLEMS

3.10.1 Users Do Not Receive Console Output

SYMPTOM   One or more users do not receive console output, and the system does not respond to keyboard input.

CAUSE   The device driver for the user's terminal was configured with EIOS buffers.

SOLUTION   Always set Max Buffers (MB) to zero on the Device-Unit Information screen when configuring a driver for a terminal.

CAUSE   An error occurred when the system tried to bring up the users.

SOLUTION   Run the INITSTATUS command from the resident user console to determine the initialization status of the users. Refer to the *Extended iRMX II.3 Operator's Guide To The Human Interface Reference Manual* for information on the INITSTATUS command.

CAUSE   The user does not have the proper access rights to some critical directory or file.

SOLUTION   Check the user's default directory (:$:), the :PROG: directory and the :CONFIG:TERMINALS file to ensure that each user has the proper access. Use the PERMIT command to change the access rights of the directory or file.

CAUSE   Either the :CONFIG:TERMINALS or :CONFIG:USER/<user-id> files are incorrect.

SOLUTION   Examine these two files for proper content and reboot. Refer to Chapter 20 in the *Extended iRMX II.3 Hardware And Software Installation Guide.*

CAUSE   The resident user has been configured with a maximum memory size of 0FFFFH. Hence, the resident user takes all available memory.

SOLUTION   Reset the resident user's memory requirements in the ICU. Ensure that sufficient memory is allocated to the non-resident users.

3.10.2 Memory-Related Error Messages

SYMPTOM A user receives a memory-related error message (for example, E$MEM) when trying to run utilities or the SUPER command in a multiuser system.

CAUSE Insufficient memory has been allocated for that user in the :CONFIG:TERMINALS file.

SOLUTION Refer to the Non-Resident Configuration Files Chapter of the *Extended iRMX II.3 Installation and Configuration Guide* for the memory partition size requirements for languages and utilities.

3.10.3 Terminals Lose Characters

SYMPTOM In a multiuser environment, a terminal may lose characters on input.

CAUSE The baud rate is too high.

SOLUTION Lower the baud rate and/or reduce the number of users.

3.11 DEBUGGING PROBLEMS

3.11.1 Breakpoints Ignored

SYMPTOM Breakpoints set in ROM with the System Debugger (SDB) are illegal (not supported).

CAUSE When a breakpoint is set, a software interrupt instruction is actually placed at the address specified. This instruction causes a jump to the monitor. These instructions cannot be written into ROM.

SOLUTION Only in-circuit emulators allow stopping normal processing at a given ROM address. Either put the code in RAM for debugging purposes or use one of the in-circuit emulators.

3.11.2 Job Flags Field Is Incorrect When Viewing A Job Token

SYMPTOM When you view a job token with the SDB command VT, the Job Flags field is incorrect.

CAUSE An error in the SDB.

SOLUTION A value of 02 for Job Flags means that parameter validation is enabeled and a value of 00 means that it is disabeled. Will be fixed in a future release.

3.11.3 Problem With The SDB Command VT

SYMPTOM If you look at a job token with the SDB command VT, an incorrect value is returned for the JOB$FLAGS field.

CAUSE Problem with the SDB.

SOLUTION This problem is fixed in Update 1 by class A ZAP ZXA033.RMD.

3.11.4 Problem With The System Debugger VT Command

SYMPTOM   The System Debugger VT command does not display the object and cache queues for a mailbox.

CAUSE   Problem with the System Debugger.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA032. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.11.5 Problem With The System Debugger's VT Concerning Buffer Pools

SYMPTOM   The System Debugger's VT command does not display any information about Data Chaining, when displaying information concerning buffer pools.

CAUSE   Problem with the System Debugger.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA023. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.11.6 You Cannot Use The "VMI" System Debugger Command

SYMPTOM   You cannot display the contents of the MPC input message buffer using the System Debugger.

CAUSE   This SDB feature was not available before Update 2 of iRMX II.3 Operating System.

SOLUTION   This feature was added in Update 2 by class A ZAP ZXA061.

3.11.7 You Cannot Use The "VMO" System Debugger Command

SYMPTOM   You cannot display the contents of the MPC output message buffer using the System Debugger.

CAUSE   This SDB feature was not available before Update 2 of iRMX II.3 Operating System.

SOLUTION   This feature was added in Update 2 by class A ZAP ZXA061.

3.11.8 You Cannot Use The "VMF" System Debugger Command

SYMPTOM   You cannot toggle the MPC fail-safe timer using the System Debugger (SDB).

CAUSE   This SDB feature was not available before Update 2 of iRMX II.3 Operating System.

SOLUTION   This feature was added in Update 2 by class A ZAP ZXA061.

3.11.9 The SDB "VT" Command Displays Erroneous Information

SYMPTOM   The System Debugger (SDB) VT command displays the device size of connection objects as a WORD instead of a DWORD.

CAUSE   Error in the System Debugger (SDB).

SOLUTION   This problem was fixed in Update 2 by class A ZAP ZXA062.

3.11.10   Using The SDB As The Default Exception Handler in iRMX II Causes GP13

SYMPTOM   When an exceptional condition occurs and execution transfers to the System Debugger, attempting to continue execution using the debugger's "GO" command causes a General Protection error.

CAUSE   The routine which calls the debugger fails to correctly handle the stack upon return.

SOLUTION   If you need to use the System Debugger as the default exception handler, answer the DEH prompt on the Nucleus ICU screen with User, which specifies a user written exception handler. Answer the NEH prompt on the Nucleus ICU screen with the name of an object file produced by assembling the following code:

```
            NAME       Exception Handler

CODE        SEGMENT    ER PUBLIC

            PUBLIC     RQSYSEX

RQSYSEX     PROC       FAR

            INT        3              ;Invoke the SDB via the SDM

            RET        8              ;Pop parameters off the stack

RXSYSEX     ENDP

CODE        ENDS

            END
```

3.11.11   A Backslash Can Hang The System When In The Monitor

SYMPTOM   A single backslash will hang the system when using the System Debug Monitor.

CAUSE   Problem in the System Debug Monitor.

SOLUTION   Reset the system. This problem will be fixed in a future update/release of the System Debug Monitor.

3.12   PATHNAME PROBLEMS

3.12.1   DIR Does Not List All Files

SYMPTOM   Some file names are not listed when using the Directory (DIR) command.

CAUSE   Files with "R?" or "r?" in their filenames are considered invisible. This is a feature of the operating system.

SOLUTION   Use the "Invisible" option of the DIR command to list these files, (that is, DIR :$: I).

3.12.2   Logical Name Does Not Exist

SYMPTOM   Some Human Interface commands do not function, usually generating the error "LOGICAL NAME DOES NOT EXIST".

CAUSE   Some commands assume the system disk device name is :SD:.

SOLUTION   Always configure systems with :SD: as the logical name for the system device.

3.12.3 DIR / Does Not List :SD:/

SYMPTOM   The "DIR /" command does not list the ":SD:/" directory.

CAUSE   When you change the default directory (:$:) to a different physical disk device with the ATTACHFILE command, the root directory (/) changes. The root directory (/) always refers to the root of the device being used as the default.

SOLUTION   Make specific references to :SD: to access the system device root directory.

3.12.4 Error When Attempt To Access File Or Directory

SYMPTOM   Attempting file or directory access causes errors.

CAUSE   Access has not been granted to the user ID attempting to use a file or directory.

SOLUTION   Determine the access rights with the EXTENDED option on the Directory (DIR) command. The file's owner can change the access rights by invoking the PERMIT command. Also, the access rights to directories, as well as files, should be verified.

3.12.5 Unauthorized Changes To Files Or Directories

SYMPTOM   Files and directories are being changed without the user's knowledge and/or access rights to files do not seem to work.

CAUSE   There are multiple users with the same user ID.

SOLUTION   File security can be enhanced by assigning different user IDs and default directories.

3.12.6 Filenames With A "?"

SYMPTOM   Problems arise when invoking the COPY, DELETE, or RENAME command on files with a "?" embedded in their filenames, (for example, R?LOGON).

CAUSE   A "?" is seen as a wildcard, not a filename character.

SOLUTION   Enclose this portion of the pathname in quotes, for example, COPY R"?"logon. This causes the enclosed characters to be taken literally.

3.13 HUMAN INTERFACE PROBLEMS

3.13.1 System Hangs When Attempting To Access Line Printer

SYMPTOM   An attempt to access a line printer that is either physically disconnected or is not turned on, hangs the system.

CAUSE   The driver waits forever for the interrupt from the printer.

SOLUTION   Do not use the line printer when it is disconnected or powered off. The workaround is press CONTROL-C, make the printer ready and re-issue the command.

3.13.2 System Hangs When iSBX 351 Terminal Is Turned Off

SYMPTOM   If a user using an iSBX 351 serial channel disconnects or turns off the terminal, the system will hang.

CAUSE   The iSBX 351 board bombards the iRMX 286 system with interrupts when in this condition.

SOLUTION   Do not turn off the terminal. This problem will be fixed in a future update or release.

3.13.3 Error Message "delete access required"

SYMPTOM   Incorrect error message (delete access required) returned by RENAME and COPY command.

CAUSE   The file's directory does not have proper access.

SOLUTION   Give the file's directory delete and add permission (using the PERMIT command) for the user executing the RENAME or COPY command.

3.13.4 Cannot Boot Off A Floppy Diskette

SYMPTOM   A bootable diskette has no errors from the DISKVERIFY command but the diskette will not boot.

CAUSE   DISKVERIFY does not verify the bootstrap loader portion of the diskette.

SOLUTION   Format the disk using the BOOTSTRAP option.

3.13.5 Unpredictable System Behavior

SYMPTOM   Unpredictable system operation.

CAUSE   A Human Interface command is not terminated properly. Failure to correctly terminate a command causes unknown code to be executed and may cause serious loss of data.

SOLUTION   Every HI command must end with either an EIOS call "RQ$EXIT$IO$JOB" or a UDI call "DQ_EXIT". The purpose of these calls is to correctly transfer control back to the iRMX II Operating System.

3.13.6 RQCSENDCORESPONSE Or RQCSENDEORESPONSE

SYMPTOM   Application program does not wait for response to RQCSENDCORESPONSE or RQCSENDEORESPONSE.

CAUSE   The Human Interface includes the string length of the response string in the response$max parameter. But, if the response$max is one, the Human Interface returns without a response$string.

SOLUTION   The value in response$max must equal the length of the string plus one.

3.13.7 Console Commands Not Executed

SYMPTOM Commands invoked from the console appear not to execute, and either the Human Interface prompt or a memory-related error message is returned.

CAUSE The size of the memory pool for the command is insufficient.

SOLUTION Increase the size of the program's memory pool in the BND286 statement. Set Max to 0FFFFFH so that it can borrow memory.

CAUSE The user has insufficient memory to satisfy the memory pool requirements of the command.

SOLUTION Enlarge the memory partition size in the :CONFIG:TERMINALS file.

3.13.8 System Time Is Incorrect After A Reboot

SYMPTOM System time and date stamp are not updated.

CAUSE The BIOS does not update the access time of any file until it is modified. The Human Interface uses the last access time of the system directory (:SYSTEM:) as the default time and date.

SOLUTION Always set the time and date when you boot your system or run shutdown before resetting or powering down your system.

3.13.9 Terminal Acts Peculiar When Running RESTORE

SYMPTOM Terminal acts peculiar when running the RESTORE command.

CAUSE RESTORE sends a 012H (DC2) character to :CO: before printing out the message "Directory Restored".

SOLUTION Run RESTORE from a submit file. This problem will be fixed in a future update or release.

3.13.10 Output Redirection To A Line Printer

SYMPTOM Redirecting output to a line printer by using the Command Line Interpreter (CLI) I/O redirection arrow (>) does not work.

CAUSE A problem with the CLI.

SOLUTION This will be fixed in a future update or release.

3.13.11 CLI I/O Redirection In An Alias

SYMPTOM CLI I/O redirection cannot be incorporated in an alias.

CAUSE A design limitation of the ALIAS command.

SOLUTION Do not use CLI I/O redirection in an alias.

3.13.12 Tapes Generated On iRMX II R2.0 Operating System Cannot Be Read On iRMX 86 Operating System

SYMPTOM   Tapes generated on an iRMX II R2.0 system can not be read on an iRMX 86 R7.0 system.

CAUSE   The granularity of the tape drive for iRMX 286 R2.0 is 7e00H.

SOLUTION   Create a tape DUIB on your iRMX 86 system with a granularity of 7e00H.

3.13.13 Continuation Lines Do Not Work In CLI Commands

SYMPTOM   A E$CONTINUATION exception occurs when using continuation lines with a CLI command.

CAUSE   CLI commands do not support continuation lines.

SOLUTION   Do not use continuation lines with CLI commands.

3.13.14 DELETE Command Executed From A Submit File Acts Peculiar

SYMPTOM   Executing a DELETE command from a submit file, with the DELETE command consisting of continuation lines and more than 256 characters, it appears that the first 256 characters after the DELETE command are lost.

CAUSE   A problem with DELETE command.

SOLUTION   Do not use the DELETE command with more than 256 characters as input. This problem will be fixed in a future update or release.

3.13.15 SHUTDOWN Hangs The System

SYMPTOM   The system hangs if you have a maximum priority of 0 for your user and you run the SHUTDOWN command.

CAUSE   An error in the SHUTDOWN command.

SOLUTION   Change in user priority.

3.13.16 Problem With The Human Interface RQCGET$PARAMETER System Call

SYMPTOM   The value of the "more" parameter of the RQCGET$PARAMETER system call is always set to zero.

CAUSE   The predict list parameter of this system call is not in CAPS.

SOLUTION   The predict list should always be in CAPS.

3.13.17 Problem With The Human Interface FORMAT Command

SYMPTOM   While formatting a disk, incorrect blocks are marked as bad blocks if an IONOSPARES error code is retured from the format process.

CAUSE   Problem with the Human Interface FORMAT command.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA028. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.13.18 DIVIDE-BY-ZERO Error Returned From The Human Interface FORMAT Command

SYMPTOM   If you use the BOOTSTRAP option of the Human Interface FORMAT command on a write-protected media you will get a DIVIDE-BY-ZERO error message.

CAUSE   Problem with the Human Interface FORMAT command.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA028. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.13.19 Cannot Use The GETBADTRACKS Option Of The FORMAT Command

SYMPTOM   No GETBADTRACKS option available for the Human Interface FORMAT command.

CAUSE   Problem with the Human Interface FORMAT command.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA028. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.13.20 Cannot Use The SETBADTRACKS Option Of The FORMAT Command

SYMPTOM   No SETBADTRACKS option available for the Human Interface FORMAT command.

CAUSE   Problem with the Human Interface FORMAT command.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA028. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.13.21 The COPY Command Does Not Return Error Messages

SYMPTOM   The COPY command does not return an error message when run programatically.

CAUSE   Problem with the COPY command.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA024. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.13.22 The DELETE Command Does Not Return Error Messages

SYMPTOM   The DELETE command does not return an error message when run programatically.

CAUSE   Problem with the DELETE command.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA025. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.13.23 The RENAME Command Does Not Return Error Messages

SYMPTOM   The RENAME command does not return an error message when run programatically.

CAUSE   Problem with the RENAME command.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA026. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.13.24 The DIR Command Does Not Return Error Messages

SYMPTOM   The DIR command does not return an error message when run programatically.

CAUSE   Problem with the DIR command.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA027. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.13.25 Problem With The Human Interface DIR Command

SYMPTOM   If you enter "DIR * SORT", when you are three or more levels deep in the file structure, the DIR command will fail to display any information.

CAUSE   Problem with the Human Interface DIR command.

SOLUTION   Enter a CONTROL-C to get to the Human Interface. This problem will be fixed in a future Update or Release.

3.13.26 Receiving NDP Exceptions When Running Programs Under Human Interface

SYMPTOM   You may encounter NDP exceptions when you run a program from the Human interface.

CAUSE   The NPX parameter in the HI JOBS screen is set to NO.

SOLUTION   The standard ICU definition files 286100A.DEF and 386100.DEF have the NPX parameter set to YES on the hardware screen, but set to NO on the HI JOBS screen. Therefore, if you have commands that will be using floating point instructions and will be run from the Human Interface you must use the ICU286 utility to set the NPX parameter in the HI JOBS screen to YES.

3.13.27 Problems Encountered When Using The MKSYS Submit File

SYMPTOM   The MKSYS submit fails during system generation.

CAUSE   Not enough memory available for PLM286 V3.1 when run from the MKSYS submit file.

SOLUTION   Increase your minimum user memory partition size to 420K bytes.

3.13.28 The "SET ALIASTABLE" CLI Command May Cause A General Protection Exception

SYMPTOM   The "SET ALIASTABLE" CLI command may cause a General Protection exception when executed from the ALIAS.CSD file.

CAUSE   The size of the ALIAS.CSD file is greater than the value specified during the "SET ALIASTABLE" command. As a workaround, you must increase the alias table size.

SOLUTION   This problem will be fixed in a future update or release.

3.13.29 The Configurable Sign-on Message Is Not Displayed

SYMPTOM   The configurable sign-on message does not get displayed during the log-in process.

CAUSE   Problem under investigation.

SOLUTION   This problem will be fixed in a future update or release.

3.13.30 You Cannot Use The Human Interface "CONNECT" Command

SYMPTOM   The iRMX II Human Interface does not recognize the "CONNECT" command.

CAUSE   This new Human Interface command was not available before Update 2.

SOLUTION   This feature was added in Update 2 by class A ZAPs ZXA012, ZXA045, ZXA046, and ZXA048.

3.13.31 You Cannot Use The Human Interface "DISCONNECT" Command

SYMPTOM   The iRMX II Human Interface does not recognize the "DISCONNECT" command.

CAUSE   This new Human Interface command was not available before Update 2.

SOLUTION   This feature was added in Update 2 by class A ZAPs ZXA012, ZXA045, ZXA046, and ZXA048.

3.13.32 The System May Hang If You Use The "JOBDELETE" Command

SYMPTOM   The system may hang if you enter the Human Interface "JOBDELETE" command.

CAUSE   The terminal being used (i.e, T0) was LOCKED and then the Command Line Interpreter job was deleted.

SOLUTION   Do not lock the console.

3.13.33 Your Response Time To HI Commands Seems Too Long

SYMPTOM   Your response time to HI commands seem too long.

CAUSE   The default priority of the SUPER user is 141. Users, other than SUPER, are created with a priority of 142. If SUPER user is active in a multi-user system, other users may notice a degradation in response time.

SOLUTION   Edit the :CONFIG:USER/SUPER file and change the priority to 142. Logout and then login again.

3.13.34 Problem With The HI RQCSENDCORESPONSE System Call

SYMPTOM   A task cannot output to the console when the Human Interface RQCSENDCORESPONSE system call has been executed and the system is waiting for user input.

CAUSE   Problem in the iRMX II Human Interface.

SOLUTION   This problem will be fixed in a future update/release of the iRMX II product.

3.13.35 An E$SHARE May Be Encountered When Using The "TIME" Command

SYMPTOM   An E$SHARE may be encountered if the Human Interface "TIME" commmmand is executed with the "GLOBAL" option.

CAUSE   The user attempted to access the global time-of-day clock while it was in use.

SOLUTION   Retry the command.

3.13.36 Problem With The Human Interface "SET" Command

SYMPTOM   AN E$SLOT message may be encountered when you execute any Human Interface command.

CAUSE   The "SET TERMINAL=VT100" command creates a segment, but does not delete it.

SOLUTION   The system must be reset. This problem will be fixed in a future update/release of the iRMX II product.

3.13.37 An E$IOMEM Exception May Be Encountered When Doing An "ATTACHDEVICE"

SYMPTOM   An E$IOMEM exception may be encountered when doing an "ATTACHDEVICE".

CAUSE   The user has not received the requested amount of memory during the "LOGON" process and therefore has insufficient memory.

SOLUTION   Ensure that the "minimum partition-size" parameter of the user attribute file (:config:user/<user-name>) is at least 512 and this value is less than the "maximum partition-size" parameter value.

3.13.38 "COPYDIR" May Hang The System

SYMPTOM   The "COPYDIR" utility may hang the system when trying to copy :SD: to a second drive.

CAUSE   Problem with the "COPYDIR" command.

SOLUTION   This problem will be fixed in a future update/release of the iRMX II product.

3.13.39 Backup Too Slow

SYMPTOM   The "BACKUP" command is extremely slow when a second tape is needed to finish the "BACKUP" process.

CAUSE   Problem with the "BACKUP" command.

SOLUTION   This problem will be fixed in a future update/release of the iRMX II product.

3.14 UTILITY PROBLEMS

3.14.1 LIB286 Truncates An Object File

SYMPTOM   When creating a new library, LIB286 truncates an object file and issues an error message "INVALID OBJECT FILE".

CAUSE   The disk has overflowed.

SOLUTION   Make sure sufficient disk space is available before you invoke LIB286.

3.14.2 LIB286 Truncates Updated File - Leaves Backup File Empty

SYMPTOM When you are updating a library, LIB286 truncates the updated file and leaves the backup file empty. LIB286 issues an error message "INVALID OBJECT FILE".

CAUSE The disk has overflowed.

SOLUTION Make sure sufficient disk space is available before you invoke LIB286.

3.14.3 Utility Generates "SYSTEM INTERFACE ERROR"

SYMPTOM The BLD286, BND286, MAP286, or OVL286 utilities generate a "SYSTEM INTERFACE ERROR". Control then transfers to the iSDM 286 Monitor, which in turn issues a stack protection error. In addition, all the other tasks running under the iRMX II multi-tasking system are halted.

CAUSE The disk has overflowed.

SOLUTION Make sure sufficient disk space is available before you invoke the utility.

3.14.4 PL/M-286 Or ASM286 Creates An Empty Object Module

SYMPTOM The PL/M-286 compiler or the ASM286 assembler appears to finish processing but creates an empty object module. No error messages appear at your terminal; however, the appropriate error messages do accumulate in the list or error print files.

CAUSE The disk has overflowed.

SOLUTION Make sure sufficient disk space is available before you invoke PL/M-286 or ASM286.

3.14.5 BND286 Truncates Input File To Length Zero

SYMPTOM The invocation sequence ":LANG:BND286 fname.lnk noload", will cause the input file "fname.lnk" to be truncated to length zero.

CAUSE The default output file created by BND286 consists of the input filename without the extension plus the extension ".lnk". So BND286 will create the output file "fname.lnk". The output filename is the same as the input filename.

SOLUTION Specify an output filename which is different from the input filename by the OBJECT control of BND286.

3.14.6 User-Written Exception Handler Does Not Receive Control

SYMPTOM A user-written exception handler does not receive control when an error occurs.

CAUSE The exception handler was compiled using the COMPACT compiler control.

SOLUTION Compile all exception handlers using the LARGE compiler control.

NOTE

All exception handlers should be in LARGE (FAR procedures) because they are called directly from the operating system.

3.14.7 Error When Invoking SUPER From AEDIT

SYMPTOM When you invoke the SUPER command from the AEDIT V2.1 editor, the exception 034H:E$CONN$NOT$OPEN is displayed.

CAUSE Error in the AEDIT V2.1 editor.

SOLUTION Be in SUPER before invoking the AEDIT V2.1 editor.

3.14.8 BND286 V3.1 Issues Error 140

SYMPTOM When using the NODEBUG control, BND286 V3.1 issues an error message "ERROR 140: TEXT LENGTH IS GREATER THAN SEGMENT LENGTH". This condition occurs with successive invocation of BND286 on the same file with a mix of DEBUG and NODEBUG controls.

CAUSE Error in BND286 V3.1.

SOLUTION Use the same control, either DEBUG or NODEBUG, in all the successive BND286 invocations.

3.14.9 BND286 V3.1 Gets General Protection Error

SYMPTOM BND286 V3.1 gets a General Protection error when binding an object module, whose assembly language source uses the ORG directive.

CAUSE To be determined.

SOLUTION Do not use ORG directives in modules assembled and bound under the iRMX 286 operating system.

3.14.10 Wrong Or Missing Version Numbers In iRMX 286-Based Languages/Tools

SYMPTOM When running the VERSION command on the iRMX 286-based languages and tools, a wrong version number or no version number is returned. The following list shows the values returned :

Name of Lang. or Tool	Value Displayed by VERSION
ASM286 V1.3	No version number found
BLD286 V3.1	X207
BND286 V3.1	X207
LIB286 V2.0	X207
MAP286 V2.1	X207
OVL286 V1.1	X207
ASM86 V2.0	No version number found
CREF86 V1.0	No version number found
LINK86 V2.7	No version number found
LOC86 V2.5	No version number found
OH86 V1.0	No version number found

CAUSE Error in each of the languages and tools.

SOLUTION This problem will be fixed in future releases of the languages and tools.

3.14.11 Problem With The 86-based ICU286 Utility

SYMPTOM   The 86-based ICU286 utility fails.

CAUSE   Problem with the 86-based ICU286 utility.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA034. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

3.14.12 The UDS Utility May Abort Prematurely

SYMPTOM   The User Device Utility (UDS) utility may abort prematurely.

CAUSE   Problem in the UDS utility.

SOLUTION   This problem was fixed in Update 2 by class A ZAPs ZXA050, ZXA051, and ZXA052.

3.14.13 Incorrect Unit Data In The .TPL File

SYMPTOM   Incorrect unit information was placed in the .TPL file when using the User Device Support (UDS) utility.

CAUSE   Problem in the UDS Utility supplied with Intel's System 120 product.

SOLUTION   This problem was fixed in Update 2 by class A ZAPs ZXA050, ZXA051, and ZXA052.

3.14.14 Problem With SCS "FETCH" Command

SYMPTOM   You cannot fetch an earlier version of an SCS file for editing purposes.

CAUSE   Problem with the SCS "FETCH" command.

SOLUTION   This problem will be fixed in a future update/release of the SCS product.

3.14.15 Error 75 Encountered When Using The SCS "FETCH" Command

SYMPTOM   Error 75 is encountered when you try to "FETCH" and edit an earlier version of an SCS file.

CAUSE   Problem with the SCS "FETCH" command.

SOLUTION   This problem will be fixed in a future update/release of the SCS product.

3.14.16 Error 16 Encountered When Using The SCS Product

SYMPTOM   Error 16 is encountered when using the SCS product.

CAUSE   SCS has problems dealing with filenames of length 12 characters or more.

SOLUTION   Make sure SCS files have 12 or less characters.

3.14.17 Error 21 Encountered When Using The SCS "GENERATE" Command

SYMPTOM Error 21 is encountered when using the SCS "GENERATE" command.

CAUSE A general dependency may have been specified twice in a General Dependency File.

SOLUTION Edit the General Dependency File and delete the general dependency that is specified twice.

3.14.18 Problem With The SCS "INFORM" Command

SYMPTOM The following SCS command will not return control to the user: INFORM file-name1 > file-name2

CAUSE The SCS product does not recognize the redirection character ">".

SOLUTION As a workaround, use the "TO" preposition instead of the redirection character ">".

3.14.19 The SCS Key Macro "Q" Is Not Recognized

SYMPTOM The SCS "INFORM" command does not recognize the key macro "Q".

CAUSE Problem with the SCS "INFORM" command.

SOLUTION This problem will be fixed in a future update or release of the SCS product.

3.14.20 General Protection Exception Encountered When Using The SCS "GENERATE" Command

SYMPTOM A General Protection exception is encountered when using the SCS "GENERATE" command.

CAUSE The number of general dependencies in the General Dependency File exceeded 64.

SOLUTION This problem has been fixed. U.S. customers may contact TIPS. European customers customers may contact their local sales office.

3.14.21 Arrays Of Length 64Kbytes And Soft-Scope 286

SYMPTOM Soft-Scope 286 cannot display the contents of an array whose size is 64Kbytes or larger.

CAUSE Problem with Soft-Scope 286.

SOLUTION This problem will be fixed in a future release/update of the Soft-Scope product.

3.14.22 Soft-Scope 286 And Local Variables In Recursive Functions

SYMPTOM Soft-Scope 286 does not recognize local variables in recursive functions.

CAUSE Problem with the Soft-Scope 286 product.

SOLUTION This problem will be fixed in a future update/release of the Soft-Scope 286 product.

3.14.23 Cannot Use The iRMX Virtual Terminal

SYMPTOM   If a user invokes VT and logs on to a remote iRMX II.3 system, does some work, logs out, and then terminates the VT session (via "~ ."), he is unable to VT to that system again.

CAUSE   Problem with Update 2 of the iRMX II.3 Operating System.

SOLUTION   As a workaround, ensure that you terminate the Virtual Terminal session (via "~ .") before you log off the system. This problem has been fixed by Class B zap ZAP076. U.S. customers may contact TIPS. European customers may contact their local sales office.

iRMX® II OPERATING SYSTEM R4.0 **4**

This chapter lists known problems related to the iRMX II Operating System, Release 4.0. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

The structure of the chapter is as follows:

4.1 System Booting Problems

4.2 Hardware Exception Problems

4.3 Application Loader Problems

4.4 Bootstrap Loader Problems

4.5 I/O System, Device, and Driver Problems

4.6 Nucleus Problems

4.7 Universal Development Interface (UDI) Problems

4.8 Interactive Configuration Utility (ICU) Problems

4.9 Mass Storage Problems

4.10 Multiuser Problems

4.11 Debugging Problems

4.12 Pathnames Problems

4.13 Human Interface Problems

4.14 Utility Problems

4.1 SYSTEM BOOTING PROBLEMS

4.1.1 Installation System Does Not Boot

SYMPTOM   The system hangs right after the sign-on message, or never comes up, while installing the system on iAPX 286-based systems.

CAUSE   This is a problem with the iSBC™ 215G firmware Version 2.3 (Part Number 146276-001 and 146276-002).

SOLUTION   Use iSBC™ 215G firmware Version 2.4 (Part Number 146979-001 and 146979-002) or more current versions. Contact your local sales office for assistance.

CAUSE   You do not have the right stepping of the iAPX 286 processor.

SOLUTION   Use at least a B-2 stepping iAPX 286 processor. Contact your local sales office for assistance.

4.1.2 System Will Not Reboot With iSBC 215G Controller

SYMPTOM   A system that worked previously now fails to initialize when an iSBC 215G controller was added to the system.

CAUSE   The iSBC 215G controller uses a wakeup address in memory that corresponds to the wakeup port address jumpered on the iSBC 215G board. This address may be inadvertently written over by the system.

SOLUTION   Be sure your wakeup address is not included in the memory specified for the system or for the free space manager by the memory screen in the ICU. For example, the default wakeup port address for the iSBC 215G board is 100H, the memory address used is 1000H through 1100H and must be excluded from the available memory in the ICU.

4.1.3 System With Multiple iSBC 215G Boards Hangs

SYMPTOM   You have two or more iSBC™ 215G boards and when you attach a disk, the system hangs.

CAUSE   The wakeup port addresses are not a minimum of 11H apart so the 272 bytes required for each board have not been reserved.

SOLUTION   If the first board has wakeup port address of 100H the second one should have 111H and the third one should have 122H etc.

4.1.4 The System Fails To Boot When Using The 8251A Device Driver

SYMPTOM   The system does not boot when the :config:terminals file assigns a Human Interface user to a terminal attached to an iSBX 351 module.

CAUSE   To be determined.

SOLUTION   Mount the iSBX 351 board or, if the iSBX 351 module is not mounted, remove the terminal attached to the iSBX 351 module from the :config:terminals file. This problem will be fixed in a future update.

4.1.5 Using The iSBC xxxCX iLBX™ RAM Boards

SYMPTOM   If your system contains more than 896 Kbytes of physical memory and you are using an iSBC xxxCX RAM Board (e.g., iSBC™ 010CX, iSBC™ 012CX, etc.), the system hangs.

CAUSE   The starting address of the Human Interface user partition lies near the 896 Kbyte boundary.

SOLUTION   You must configure your system so that no user partition begins within 50 Kbytes of the 896 Kbyte boundary.

4.1.6 Baud Rate Scan Of Terminal Set To 9600 Fails

SYMPTOM   A baud rate scan of a terminal set to 9600 baud causes the system to fail, if your system contains either an iSBC board with clockrate greater than 6MHz (such as the iSBC 286/10A board or the iSBC 286/12 board) or an iSBC board that utilizes 1 or 0 wait-state memory and uses an 8274 Terminal Driver.

CAUSE   Timing problem in interfacing with the 8274 MPC.

SOLUTION   Use two or more wait-states in memory or set the terminal baud rate to something other than 9600 baud when doing the baud rate scan or configure the 8274 Terminal Driver to use a fixed baud rate.

4.1.7 "no-go" From Winchester

SYMPTOM   When an iRMX system box is powered up, the SCT shows a "no-go" for the Winchester disk, but subsequent boots while system is powered up show a "go" for the disk.

CAUSE   If your system has "autolock" for the disk, the SCT will fail on power up for the disk because the delay is not long enough to allow for the auto-unlock. The term "autolock" means that when the power is turned off, the heads are automatically locked, versus manual lock where a switch had to be manually set to lock the heads.

SOLUTION   A future release of the SCT will have a longer delay.

4.1.8 No Response To Keyboard - No Output To Console

SYMPTOM   When you attempt to boot the system, it accesses the disk, but neither responds to keyboard input nor sends any output to the console.

CAUSE   The third stage Bootstrap Loader is in the same memory that is reserved for the second stage of the Bootstrap Loader.

SOLUTION   If you configure a third stage Bootstrap Loader, do not locate it in the space allocated for the second stage of the Bootstrap Loader. See table.

SCT Version Number	Starting RAM Location	Size of RAM Area (in bytes)
SCT 286/300x (all versions)	24K bytes less than the largest RAM address (not to exceed 0C0000H)	21K (5400H)

CAUSE   The initial task belonging to a first-level job has a missing or unexecuted END$INIT$TASK system call.

SOLUTION   Check that the initial task of all first-level user jobs invokes an END$INIT$TASK system call.

CAUSE   The :CONFIG:TERMINALS file has been corrupted, or has zero users allocated.

SOLUTION Create or correct the :CONFIG:TERMINALS file.

CAUSE The :CONFIG:USER/<user id> files are incorrect. A common error is to use both a comma (,) and a carriage return as a parameter delimiter. If a comma is followed by a carriage return, a null parameter is assigned.

SOLUTION Correct the :CONFIG:USER/<user id> files to ensure that commas are not unintentionally followed by carriage returns.

CAUSE Insufficient memory to complete the booting process.

SOLUTION Check that your system has enough memory to complete booting.

CAUSE Memory-mapped I/O addresses were not excluded from the S and FSM memory.

SOLUTION Exclude memory-mapped I/O addresses from iRMX II memory using the MEMORY FOR SYSTEM screen of the ICU.

CAUSE The system has been configured to use the System Debug Monitor (iSDM™), but the iSDM monitor has not been physically installed in the system or it has been placed at an invalid address.

SOLUTION Install the iSDM monitor. For information about how to install and/or use the iSDM monitor, refer to the *iSDM™ System Debug Monitor Installation and Configuration Guide.*

CAUSE The system device name in the Bootstrap Loader does not have a corresponding device name in a Device Unit Information entry within the system definition file.

SOLUTION Using the ICU, make sure that the device name of the booting device exists in a Device Unit Information entry. For example, "W" for the Winchester disk.

CAUSE There is a conflict over an interrupt. For example, two or more interrupts are assigned to the same level.

SOLUTION Check the board jumpers and the configuration definition file for two interrupts assigned to the same level.

CAUSE The interrupt from a device is not getting to the 8259A Programmable Interrupt Controller.

NOTE

The Bootstrap Loader does not use interrupts.

SOLUTION Check that the interrupt is actually getting to the 8259A (PIC). Refer to the hardware reference manual for your Single Board Computer (iSBC).

CAUSE The disk has been formatted using the iRMX 86 Release 7 FORMAT and the second stage of the bootstrap loader for the iRMX 86 Release 7 system is not compatible with the iRMX II system.

SOLUTION Format the disk using iRMX II FORMAT.

4.1.9 ROM System Does Not Come Up

SYMPTOM When a ROM based iRMX II operating system is powered up, it does not come up.

CAUSE The iSBC 286/10A or iSBC 286/10 board is not jumpered correctly.

SOLUTION To specify four 27512 EPROM devices, set up jumpers 62 through 91 as specified in the 27512 (64K x 8 EPROM) example in Appendix E of the *iSBC 286/10A or iSBC 286/10 Hardware Reference Manual.*

To specify a starting memory address and memory size for local memory, use primary decode option 3. The jumpers required are:

Default Configuration	Jumpers for Primary Decode Option 3
E218 - E219 installed	E218 - E219 installed
E220 - E221 removed	E220 - E221 installed

To specify memory size/justification for local memory, use secondary option 3. The jumpers required are:

Default Configuration	Jumpers for Primary Decode Option 3
E51 - E59 removed	E51 - E59 removed
E50 - E58 removed	E50 - E58 installed
E49 - E57 installed	E49 - E57 installed

4.1.10 EIOS Initialization Error 40h Encountered

SYMPTOM EIOS Initialization Error 40h is encountered when bootstrap loading an iRMX II operating system.

CAUSE The cause is under investigation.

SOLUTION Re-boot your iRMX II operating system. Will be fixed in a future release.

4.1.11 Error 118 Received From BLD286 Utility

SYMPTOM The BLD286 utility generates the following error message during system configuration.

```
*** ERROR 118: Input Segments Exceed Target Memory Processing Aborted
```

CAUSE The amount of memory needed for the iRMX 286 operating system code and data segments has exceeded the amount of memory given to it during system configuration.

SOLUTION Use the ICU286 utility and increase the system memory on the memory for system screen. Decrease the memory specified on the memory for free space manager screen by the same amount. Refer to the segment description portion of the BLD286 map file (<boot_file>.mp2).

4.1.12 Checksum Error Encountered While Bootloading The Operating System

SYMPTOM A checksum error is encountered when you bootload the iRMX II operating system.

CAUSE A file was specified in the first stage of the bootstrap loader that was not the third stage of the bootstrap loader.

SOLUTION Always specify the third stage of the bootstrap loader when bootloading your system.

4.1.13 Problems With Installation Of Update 2

SYMPTOM Many error messages may be generated during the installation process of Update 2.

CAUSE The LIB86 utility is required for the installation of Update 2.

SOLUTION Make sure you have the LIB86 utility before installing Update 2.

4.1.14 Memory Warning Received During the Logon Process

SYMPTOM   The operating system may display a WARNING message during the logon process, informing the user that it cannot assign it the minimum memory it has requested.

CAUSE   The maximum memory partition-size value of the user attribute file was smaller than the minimum memory partition-size.

SOLUTION   Contents of the user attribute file before and after the problem:

Before: 2000,512,141,:sd:user/super After : 512,2000,141,:sd:user/super

4.1.15 An "INTERRUPT 3" May Be Encountered When Bootloading The Operating System

SYMPTOM   An "INTERRUPT 3" may be encountered when bootloading the iRMX II Operating System from a single high density, 96 TPI boot diskette.

CAUSE   Problem in the iRMX II bootstrap loader. As a workaround you must use the generic third stage of the bootstrap loader for your specific board.

SOLUTION   This problem will be fixed in a future update/release of the iRMX II product.

4.1.16 EIOS Initialization Error Encountered When Auto Boot Is Used In MULTIBUS II

SYMPTOM   An EIOS initialization error 40H may be encountered when bootloading automatically on a MULTIBUS II system.

CAUSE   A problem with the Human Interface FORMAT command.

SOLUTION   This problem was fixed by Class A zap ZAP041. U.S. customers must contact TIPS. European customers must contact their local sales office.

4.2 HARDWARE EXCEPTION PROBLEMS

4.2.1 General Protection In A System Call

SYMPTOM   You get a general protection error within a system call, even though the parameters passed to the call are correct.

CAUSE   The application program is bound to the wrong interface library. For example, a LARGE program is bound to a COMPACT interface library.

SOLUTION   Check your program size control statement and the bind command to make sure they are compatible.

4.2.2 General Protection In An Application Program

SYMPTOM   When running an application program, you get a general protection error.

CAUSE   You have an invalid data segment or you have crossed the boundary of the segment.

SOLUTION   Make the necessary correction to your program.

NOTE

The iSDM command "ddt(ds)" will display the data segment characteristics. By examining the data segment you can identify the problem.

CAUSE You have an invalid extra segment or you have crossed the boundary of the segment.

SOLUTION Make the necessary correction to your program.

NOTE

The iSDM command "ddt(es)" will display the extra segment characteristics. By examining the extra segment you can identify the problem.

4.2.3 Stack Protection In An Application Program

SYMPTOM When running an application program you get a stack protection error.

CAUSE You have an invalid stack segment or the stack has overflowed.

SOLUTION Make the necessary correction to your program.

NOTE

The iSDM command "ddt(ss)" will display the stack segment characteristics. By examining the stack segment you can identify the problem.

CAUSE The PL/M-286 compiler did not create a stack and the application loader did not create a stack big enough for the execution of the program.

SOLUTION Use debug control when compiling.

4.2.4 Stack Protection Error When Running BLD286

SYMPTOM When running BLD286 on a system that does not have enough disk space, a stack protection exception is encountered. The builder displays the following messages a number of times before it gets a protection error:

```
INTERFACE ERROR
0029 : E$SPACE
FILE : ROM.MP2
PROCESS ABORTED
```

CAUSE The disk does not have enough space to contain the output files from the build operation.

SOLUTION Make sure your disk has over 1 Mbyte free space before starting the ICU. This problem will be fixed in a future release of BLD286.

4.2.5 Third Stage Gets DIVIDE-BY-ZERO Error

SYMPTOM The third stage of the Bootstrap Loader gets a DIVIDE-BY-ZERO error.

CAUSE Your system was not reset before you booted your iRMX II operating system.

SOLUTION Reset the system by pressing the reset button on your system's front panel or by whatever means you have built into your hardware.

CAUSE You have configured the IDT with too few slots.

SOLUTION Check your definition file and correct the number of IDT entries in the Nucleus screen of the ICU.

CAUSE   You are using an iRMX 86 operating system Bootstrap Loader second stage. In other words, you are booting from a disk which was formatted using the iRMX 86 FORMAT command.

SOLUTION Reformat the disk using iRMX II FORMAT command with BOOTSTRAP option only the second stage of the Bootstrap Loader is written by this option; the other files on the disk are uneffected.

4.3   APPLICATION LOADER PROBLEMS

4.3.1   Wrong Message Returned To MSG$MBOX In RQ$S$LOAD$IO$JOB

SYMPTOM   When using the RQSLOADIOJOB system call, the mailbox specified in the MSG$MBOX parameter receives the wrong message.

CAUSE   The mailbox specified in the MSG$MBOX parameter of the RQ$S$LOAD$IO$JOB call is used internally by the loader to receive the loader result segment from RQALOADIOJOB. This causes several problems:

1. The user cannot specify selector$of(NIL) for the MSG$MBOX parameter as in RQ$CREATE$IO$JOB.

2. The user can receive the wrong message. If the user has a separate task to receive exit messages from I/O jobs and this task does a RQ$RECEIVE$MESSAGE at the MSG$MBOX before another task calls RQSLOADIOJOB, the user will receive the loader result segment and the system will crash.

3. The user can receive the wrong message. If the user has 2 RQSLOADIOJOB calls that specify the same MSG$MBOX, the loader can receive the exit message from the first I/O job while expecting to get the loader result segment from the second load. This prevents delivery of the exit message and causes the second load to terminate without success.

SOLUTION As a workaround, avoid the following:

1. Specify NIL for the MSG$MBOX of the RQ$S$LOAD$IO$JOB

2. Leave a task waiting for a message at a mailbox and then use the same mailbox in a RQSLOADIOJOB call. This problem will be fixed in a future update or release.

4.3.2   The Application Loader Reduces The Number Of Available GDT Slots

SYMPTOM   When loading applications containing overlays the number of available GDT slots is decreased after the application had completed execution.

CAUSE   A problem in the Application Loader.

SOLUTION This problem was fixed by Class A zaps ZAP031. U.S. customers must contact TIPS. European customers must contact their local sales office.

4.3.3   General Protection Exception Encountered In The Loader

SYMPTOM   A General Protection exception may be encountered in the Application Loader code when trying to load an OVERLAYED application.

CAUSE   A problem in the Application Loader.

SOLUTION This problem was fixed by Class A zaps ZAP035, ZAP036, and ZAP037. U.S. customers must contact TIPS. European customers must contact their local sales office.

4.3.4 "E$MEM" Exception Encountered When Running A Program

SYMPTOM   An "E$MEM" exception may be encountered when running a program under the Human Interface.

CAUSE   A problem in the Application Loader.

SOLUTION   This problem was fixed by Class A zaps ZAP035, ZAP036, and ZAP037. U.S. customers must contact TIPS. European customers must contact their local sales office.

4.4 BOOTSTRAP LOADER PROBLEMS

4.4.1 Third Stage Does Not Issue The First Message

SYMPTOM   While bootloading, the message, NOW IN PROTECTED MODE, is not issued and the system hangs.

CAUSE   The load file you specified with a .286 extension or the default load file you used, STEM/RMX86.286 (if no name is specified) was not found on the volume.

SOLUTION   Check the volume to make sure the file specified or the default file STEM/RMX86.286 exists.

CAUSE   The pathname for the third stage has a period in it other than the period for the file name extension.

SOLUTION   Rename the directory or the filename in the pathname of the third stage, so that the name does not contain a period, except the period for the extension of the third stage's file name.

For example, if the third stage has a pathname /BOOT/MY.DIR.SYS286.3RD, change it to /BOOT/MYDIR/SYS286.3RD.

4.4.2 Third Stage Does Not Issue The Second Message

SYMPTOM   While bootloading, the message, TARGET FILE LOADED, is not issued and the system hangs.

CAUSE   The file containing the system was incorrect.

SOLUTION   Check the file and regenerate the system if necessary.

CAUSE   The third stage was loading the load file on top of it.

SOLUTION   Use the memory screen of the ICU to correct your memory allocation. By default, the second stage is located at B8000H, BBFFFH - C0000H, the device generic and specific third stages are located at BC000H - C0000H. Your system memory should not include these locations. However, the free space manager can include this memory.

4.4.3 Third Stage Gets A DIVIDE-BY-ZERO Error

SYMPTOM   The third stage of the Bootstrap Loader gets a DIVIDE-BY-ZERO error.

CAUSE   Your system was not reset before you booted your iRMX 286 operating system.

SOLUTION   Reset the system by pressing the reset button on your system's front panel or by whatever means you have built into your hardware.

CAUSE   You have configured the IDT with too few slots.

SOLUTION Check your definition file and correct the number of IDT entries in the Nucleus screen of the ICU.

CAUSE You are using an iRMX 86 operating system Bootstrap Loader second stage. In other words, you are booting from a disk which was formatted using the iRMX 86 FORMAT command.

SOLUTION Reformat the disk using the iRMX 286 FORMAT command with the BOOTSTRAP option.

4.4.4 Checksum Error Encountered While Bootloading The Operating System

SYMPTOM A checksum error is encountered when you bootload the iRMX II operating system.

CAUSE A file was specified in the first stage of the bootstrap loader that was not the third stage of the bootstrap loader.

SOLUTION Always specify the third stage of the bootstrap loader when bootloading your system.

4.4.5 An "INTERRUPT 3" May Be Encountered When Bootloading The Operating System

SYMPTOM An "INTERRUPT 3" may be encountered when bootloading the iRMX II operating system from a single high density, 96 TPI boot diskette.

CAUSE Problem in the iRMX II bootstrap loader. As a workaround you must use the generic third stage of the bootstrap loader for your specific board.

SOLUTION This problem will be fixed in a future update/release of the iRMX II product.

4.5 I/O SYSTEM, DEVICE, AND DRIVER PROBLEMS

4.5.1 System Time Not Current When CONTROL-C Out Of TIME Command

SYMPTOM If you abort the default r?logon program at signon time via a CONTROL-C, the default Date/Time of the system does not automatically revert to the last time the directory :SYSTEM: was accessed.

CAUSE The system gets its time from the :SYSTEM: directory if the operator does not enter time during signon time. To improve performance, iRMX II does not update a file's access parameter unless the file was modified. Because of this, the access time of STEM: directory reflects the last time it was modified.

SOLUTION Enter time each time you logon. Run "SHUTDOWN" before you turn power off or reset your system.

4.5.2 iSBC 188/48 Board Ignores Characters With Parity Errors

SYMPTOM The iSBC 188/48 driver does not pass thru characters with input parity errors.

CAUSE A feature of the iSBC 188/48 firmware.

SOLUTION Use a different device. For example, an iSBC 544A board may be used.

4.5.3 iSBC 188/48 and iSBC 188/56 Board Problem

SYMPTOM A General Protection error is encountered in a system that uses the iSBC 188/48 or iSBC 188/56 board.

CAUSE A problem with the iSBC 188/48 and iSBC 188/56 boards (if IC U-69 is not labeled 451012).

SOLUTION Return the board to the repair center for upgrade.

4.5.4 Translation Of Escape Sequence #18 Does Not Work

SYMPTOM Translation of escape sequence 18 does not work.

CAUSE A problem in the Terminal Support Code.

SOLUTION This problem will be fixed in a future update or release.

4.5.5 Cannot Use iSBX 251 At Addresses Other Than 80H

SYMPTOM The iSBX 251 driver will not work if you move the iSBX 251 driver data port address, anywhere other than a data port address of 80H.

CAUSE A problem with the ICU where the IDEVC.A28 file does not reflect the change of data port address made when configuring the driver.

SOLUTION Edit the IDEVC.A28 file to reflect the correct data port address. This problem will be fixed in a future update or release.

4.5.6 CONTROL-S Can Hang A Terminal

SYMPTOM While scrolling through a file using AEDIT, if you enter a CONTROL-S you may not be able to resume using a CONTROL-Q.

CAUSE Problem under investigation.

SOLUTION This problem will be fixed in a future update or release.

4.5.7 Your Response Time To HI Commands Seems Too Long

SYMPTOM You response time to HI commands seem too long.

CAUSE The default priority of the SUPER user is 141. Users, other than SUPER, are created with a priority of 142. If SUPER user is active in a multi-user system, other users may notice a degradation in response time.

SOLUTION Edit the :CONFIG:USER/SUPER file and change the priority to 142. Logout and then login again.

4.5.8 General Protection Exception Encountered When Saving A File Using AEDIT

SYMPTOM A General Protection exception may be encountered when saving a file on a floppy using AEDIT.

CAUSE The floppy was removed from the diskette drive, before the save command was issued. The problem is under investigation.

SOLUTION Ensure that a floppy diskette is in the drive and the floppy drive door is closed.

4.5.9 Problem With The iSBC 186/410 Device In Full Duplex Mode

SYMPTOM   Input is extremely slow compared to output on the iSBC 186/410 device when used in full duplex mode.

CAUSE   Problem with the iSBC 186/410 device driver.

SOLUTION   This problem will be fixed in a future update/release of the iRMX II product.

4.5.10 An E$IOMEM Exception May Be Encountered When Doing An "ATTACHDEVICE"

SYMPTOM   An E$IOMEM exception may be encountered when doing an "ATTACHDEVICE".

CAUSE   The user has not received the requested amount of memory during the "LOGON" process and therefore has insufficient memory.

SOLUTION   Ensure that the "minimum partition-size" parameter of the user attribute file (:config:user/<user-name>) is at least 512 and this value is less than the "maximum partition-size" parameter value.

4.5.11 Cannot Input More Than 255 Bytes At A Time When Using The Computer-To-Computer Feature

SYMPTOM   One cannot input more than 255 bytes at a time when using the computer-to-computer feature of the iRMX II operating system.

CAUSE   This is a limitation of the iRMX II operating system.

SOLUTION   Repeated input statements will allow the program to obtain the data being sent to it.

4.5.12 The Downloader System Job Cannot Support Multiple ISBC 186/410 Boards

SYMPTOM   The Downloader System job cannot support more than one iSBC 186/410 board.

CAUSE   A problem in the Downloader System job.

SOLUTION   This problem was fixed by Class A zaps ZAP076, ZAP077, ZAP079, ZAP089, and ZAP090. U.S. customers must contact TIPS. European customers must contact their local sales office.

4.6 NUCLEUS PROBLEMS

4.6.1 System Is In Infinite Loop

SYMPTOM   If the SDB is not the exception handler in your system and the exception mode is "never", a DIVIDE-BY-ZERO error in your application program will cause an infinite loop to be executed.

CAUSE   The system default exception handler and the system alternative exception handler do not increment the IP register past the offending instruction.

SOLUTION   Use the SDB as the exception handler during the development phase and write a user exception handler if DIVIDE-BY-ZERO can occur in the application.

4.6.2 Cannot Set Exception Handlers For Hardware Exceptions

SYMPTOM RQSETEXCEPTION$HANDLER will not install a user-written exception handler for a hardware exception.

CAUSE The iRMX II Release 4.0 system does not provide this capability.

SOLUTION This problem will be fixed in a future release.

4.6.3 RQ$RESUME$TASK Call Gets E$STATE

SYMPTOM A task which makes a RQ$DISABLE$DELETION call to protect itself from deletion and then makes a RQ$SUSPEND$TASK call to suspend itself, cannot be resumed by a RQ$RESUME$TASK call. The RQ$RESUME$TASK returns a condition code of E$STATE which indicates that the task was not suspended.

CAUSE Error in NUC.LIB.

SOLUTION This problem will be fixed in a future update.

4.6.4 Exceptions Are Not Caught By The Exception Handler

SYMPTOM Environmental exceptions are not caught by the exception handler.

CAUSE The UDI exception handler acts only on programmer exceptions, and when binding your program you specified the UDI Interface Library before the iRMX 286 Interface Library.

SOLUTION In your BND286 statement specify the iRMX 286 Interface Library before the UDI Interface Library.

4.6.5 E$STATE Exception Never Returned By RQ$RESUME$TASK

SYMPTOM An E$STATE exception is not returned if you resume a non-suspended task with the Nucleus call RQ$RESUME$TASK.

CAUSE An error in the Nucleus.

SOLUTION Will be fixed in a future release.

4.6.6 Problem With The RQ$RESUME$TASK() System Call

SYMPTOM If you resume a non-suspended sleeping task with the nucleus system call RQ$RESUME$TASK, the nucleus does not return an E$STATE exception.

CAUSE Problem with the RQ$RESUME$TASK system call.

SOLUTION This problem will be fixed in a future update or release.

4.6.7 RQEOBJECT$ACCESS Returns An E$BAD$ADDRESS Exception

SYMPTOM The RQEOBJECT$ACCESS nucleus system call returns a EBADADDRESS exception if the ACCESS$PTR parameter of this call points to a structure that is located at the end of the data segment.

CAUSE Problem with the RQEOBJECT$ACCESS system call.

SOLUTION This problem will be fixed in a future update or release.

4.6.8 E$CONTEXT Received From Buffer Pool System Calls

SYMPTOM An E$CONTEXT is received when executing buffer pool system calls.

CAUSE The buffer pools and the buffers that are used with the buffer pool system calls were not created by the job they are used in.

SOLUTION Make sure that the buffer pools and the segments (buffers) that are used with the buffer pool system calls are created by the job they are used in.

4.6.9 The iRMX Operating System Does Not Support The 80287 Emulator

SYMPTOM The iRMX Operating System does not support the 80287 Emulator. Incorrect Floating Point values may be returned.

CAUSE The 80287 Emulator has non-reentrant routines. Therefore the state of the Emulator will not be saved during a task switch.

SOLUTION You must either obtain and configure an 80287 component or synchronize your tasks (i.e, using SEMAPHORES) in a way that only one task uses the Emulator at any time.

4.6.10 RQ$DELETE$BUFFER$POOL System Call Always Returns An E$OK

SYMPTOM A call to the RQ$DELETE$BUFFER$POOL Nucleus system call to delete a buffer pool that is attached to a port always returns a condition code of E$OK.

CAUSE Problem in the iRMX Nucleus.

SOLUTION This problem will be fixed in a future update or release.

4.6.11 Problem With The RQ$SEND$RSVP Nucleus System Call

SYMPTOM Control does not return from the RQ$SEND$RSVP nucleus system call.

CAUSE Existing, but incorrect "port$tkn" and "socket" parameters were used for this system call.

SOLUTION Use valid and correct "port$tkn" and "socket" parameters.

4.6.12 General Protection Exception Encountered When Using The RQ$DELETE$DESCRIPTOR Nucleus System Call

SYMPTOM A General Protection Exception may be encountered when the RQ$DELETE$DESCRIPTOR Nucleus system call is used.

CAUSE A Problem in the iRMX II Nucleus.

SOLUTION This problem will be fixed in a future update/release of the iRMX II product.

4.6.13 An E$INTERRUPT$OVERFLOW Message Encountered

SYMPTOM An E$INTERRUPT$OVERFLOW message may be encountered from the Nucleus RQ$TIMED$INTERRUPT system call.

CAUSE Problem with the iRMX II.3 Nucleus.

SOLUTION This problem will be fixed in a future update/release of the iRMX II product.

4.6.14 NUCLEUS INITIALIZATION ERROR: 0002H Encountered

SYMPTOM A "NUCLEUS INITIALIZATION ERROR : 0002H" may be encountered when bootloading the iRMX II operating system.

CAUSE All available memory was designated as system memory. Therefore, the operating system had insufficient memory for its initialization.

SOLUTION Use the interactive configuration utility, and change the system memory screen to allow sufficient memory for the system to come up and initialize.

4.6.15 A General Protection Exception May Be Encountered When Executing A RQ$RECEIVE$DATA Nucleus System Call

SYMPTOM A general protection exception may be encountered when executing an RQ$RECEIVE$DATA nucleus system call.

CAUSE Problem in iRMX II nucleus. As a workaround you must ensure that the receive buffer size is at least 128 bytes in length.

SOLUTION This problem will be fixed in a future update/release of the iRMX II product.

4.6.16 The Nucleus System Call "RQ$RECEIVE" Returns An Incorrect Flag

SYMPTOM The Nucleus system call "RQ$RECEIVE" erroneously returns a flag indicating that a returned contiguous data message buffer is a data chain.

CAUSE A problem in the Nucleus.

SOLUTION This problem was fixed by Class A zap ZAP030. U.S. customers must contact TIPS. European customers must contact their local sales office.

4.7 UNIVERSAL DEVELOPMENT INTERFACE (UDI) PROBLEMS

4.7.1 CONTROL-C Handler Gets Stack Protection Error

SYMPTOM A user-written CONTROL-C handler gets a Stack Protection error.

CAUSE The CONTROL-C handler was compiled using the COMPACT model of compilation.

SOLUTION Compile the CONTROL-C handler using the LARGE model of compilation.

4.7.2 Run Time Error 0001H Encountered In An I/O Job

SYMPTOM When making a FORTRAN written application an I/O job, you receive a run time exception of 0001H.

CAUSE A problem with the UDI.

SOLUTION Lookup "R?HCNAME" in the CLI's job and catalog it in the I/O job's directory under the same name. This problem will be fixed in a future update or release.

4.8 INTERACTIVE CONFIGURATION UTILITY (ICU) PROBLEMS

4.8.1 BLD286 Issues "REFERENCE TO EMPTY SEGMENT" Error Message

SYMPTOM   If a user module contains only large model reentrant procedures and this module is specified as a user module in the ICU, then BLD286 issues an error message "REFERENCE TO EMPTY SEGMENT".

CAUSE   Reentrant procedures use the stack for their data. As a result, the data segment remains empty. In the large model, this empty data segment does not get combined with the data segments of other modules.

SOLUTION   Declare a "dummy" variable in the module at the module level, so that the data segment is not empty.

4.8.2 Changes To Definition File Were Not Saved

SYMPTOM   After generating a system, changes made in the ICU session did not get changed in the definition file.

CAUSE   The ICU tried to access files without having the correct access rights for those files.

SOLUTION   Refer to chapter two of the *Extended iRMX II.3 Configuration Guide* for correct procedures on access rights of ICU files.

4.8.3 Receiving NDP Exceptions When Running Programs Under Human Interface

SYMPTOM   You may encounter NDP exceptions when you run a program from the Human interface.

CAUSE   The NPX parameter in the HI JOBS screen is set to NO.

SOLUTION   The standard ICU definition files 286100A.DEF and 386100.DEF have the NPX parameter set to YES on the hardware screen, but set to NO on the HI JOBS screen. Therefore, if you have commands that will be using floating point instructions and will be run from the Human Interface you must use the ICU286 utility to set the NPX parameter in the HI JOBS screen to YES.

4.8.4 Problems Encountered When Using The MS Submit

SYMPTOM   The MS submit fails during system generation using PL/M-286 V3.1 and/or ASM286 V1.3.

CAUSE   Not enough memory available for PLM286 V3.1 when run from the MS submit file.

SOLUTION   Increase your minimum user memory partition size to 420K bytes.

4.8.5 Hardware Screen Not Found When Using The ICU Utility

SYMPTOM   A warning may be encountered when you try to change the hardware parameters using the ICU utility.

CAUSE   The default "Detail-Level" has been changed in the ICU to avoid the screens not used by most users.

SOLUTION   As a workaround, you must change the "Detail-Level" to "ALL" before trying to change the hardware screen parameters.

4.8.6 General Protection Exception Encountered When Using The ICU Help Option

SYMPTOM   A General Protection exception may be encountered when using the "HELP" option on the parameters of the iRMX-NET screen with the ICU utility.

CAUSE   A problem in the ICU utility.

SOLUTION   This problem was fixed by Class A zaps ZAP071 and ZAP071. U.S. customers must contact TIPS. European customers must contact their local sales office.

4.8.7 Terminal Support Code Is Added When Not Required

SYMPTOM   The Terminal Support Code is included in a system that does not include any terminal drivers.

CAUSE   A problem in the ICU utility.

SOLUTION   This problem was fixed by Class A zaps ZAP070 and ZAP071. U.S. customers must contact TIPS. European customers must contact their local sales office.

4.8.8 AT-BUS ROM Support Screen Missing

SYMPTOM   The AT-BUS ROM support screen may be missing.

CAUSE   A problem in the ICU utility.

SOLUTION   This problem was fixed by Class A zaps ZAP070 and ZAP071. U.S. customers must contact TIPS. European customers must contact their local sales office.

4.9 MASS STORAGE PROBLEMS

4.9.1 Cannot Format iSBX 251 Multimodule

SYMPTOM   The user cannot format an iSBX 251 Magnetic Bubble Memory Multimodule.

CAUSE   Improper jumpering on iSBX 251 multimodule.

SOLUTION   On the iSBX 251 multimodule, remove jumper W5 and install jumper W6. For more information on jumpering the iSBX 251 multimodule, refer to the *iSBX™ 251 Multimodule Technical Reference Manual*.

4.9.2 Data Loss On iSBX 251 Driver

SYMPTOM   Data loss occurs on an iSBX 251 driver.

CAUSE   The CPU must service all interrupts from the iSBX 251 driver within 1.5 milliseconds. Failure to respond within this time frame could cause the iSBX 251 driver to lose data.

SOLUTION   Solutions are as follows:

1. Do not mount the iSBX 251 Multimodule on a board with a clockrate greater than 6MHz (such as an iSBC 286/12).

2. Do not mount the iSBX 251 Multimodule on a CPU board that uses less than two wait-states on the iLBX.

3. Do not use a terminal that uses the terminal support code.

4.9.3 Data Loss On Mass Storage Device

SYMPTOM Data loss occurs on a mass storage device.

CAUSE System data buffers have not been written to the storage device prior to power loss or pressing the reset button.

SOLUTION Following disk access, wait for the update timeout (usually two seconds) before pressing the reset button or powering down the system.

NOTE

It is good practice to close all files and use the system call A$UPDATE while accessing disk files under program control. When rebooting or powering down your system, always run shutdown first.

4.9.4 Data Loss On A Diskette

SYMPTOM Data loss occurs on a diskette.

CAUSE Some diskette drives do not have a Ready Line to inform the operating system when diskettes are changed; this applies to 5.25" drives, slim-line 8" drives, and micro floppy drives.

SOLUTION Use the ATTACHDEVICE command after a new diskette has been inserted and use the DETACHDEVICE command before removing a diskette from the drive.

4.9.5 Cannot Use iSBX 251 At Addresses Other Than 80H

SYMPTOM The iSBX 251 driver will not work if you move the iSBX 251 driver data port address, anywhere other than a data port address of 80H.

CAUSE A problem with the ICU where the IDEVC.A28 file does not reflect the change of data port address made when configuring the driver.

SOLUTION Edit the IDEVC.A28 file to reflect the correct data port address. This problem will be fixed in a future update or release.

4.9.6 A "VOLUME NOT MOUNTED" Message Encountered While Installing The iRMX II.4 Operating System

SYMPTOM A "VOLUME NOT MOUNTED" message may be encountered while installing the iRMX II.4 Operating System.

CAUSE This message is generated when you try to format your hard disk using the "form_disk" submit file. The command fails when the "changesd" command is executed to switch the system device from the floppy to the hard disk. This problem was caused by an incorrect jumpering of the disk drive.

SOLUTION Make sure your disk drive is jumpered correctly. For disk drive jumper information, refer to the hardware reference manual for your particular brand of drive.

4.10   MULTIUSER PROBLEMS

4.10.1   Users Do Not Receive Console Output

SYMPTOM   One or more users do not receive console output, and the system does not respond to keyboard input.

CAUSE   The device driver for the user's terminal was configured with EIOS buffers.

SOLUTION   Always set Max Buffers (MB) to zero on the Device-Unit Information screen when configuring a driver for a terminal.

CAUSE   An error occurred when the system tried to bring up the users.

SOLUTION   Run the INITSTATUS command from the resident user console to determine the initialization status of the users. Refer to the *Extended iRMX II.3 Operator's Guide To The Human Interface Reference Manual* for information on the INITSTATUS command.

CAUSE   The user does not have the proper access rights to some critical directory or file.

SOLUTION   Check the user's default directory (:$:), the :PROG: directory and the :CONFIG:TERMINALS file to ensure that each user has the proper access. Use the PERMIT command to change the access rights of the directory or file.

CAUSE   Either the :CONFIG:TERMINALS or :CONFIG:USER/<user-id> files are incorrect.

SOLUTION   Examine these two files for proper content and reboot. Refer to Chapter 20 in the *Extended iRMX II.3 Hardware And Software Installation Guide*.

CAUSE   The resident user has been configured with a maximum memory size of 0FFFFH. Hence, the resident user takes all available memory.

SOLUTION   Reset the resident user's memory requirements in the ICU. Ensure that sufficient memory is allocated to the non-resident users.

4.10.2   Memory-Related Error Messages

SYMPTOM   A user receives a memory-related error message (for example, E$MEM) when trying to run utilities or the SUPER command in a multiuser system.

CAUSE   Insufficient memory has been allocated for that user in the :CONFIG:TERMINALS file.

SOLUTION   Refer to the Non-Resident Configuration Files Chapter of the *Extended iRMX II.3 Installation and Configuration Guide* for the memory partition size requirements for languages and utilities.

4.10.3   Terminals Lose Characters

SYMPTOM   In a multiuser environment, a terminal may lose characters on input.

CAUSE   The baud rate is too high.

SOLUTION   Lower the baud rate and/or reduce the number of users.

4.11   DEBUGGING PROBLEMS

4.11.1   Breakpoints Ignored

SYMPTOM   Breakpoints set in ROM with the System Debugger (SDB) are illegal (not supported).

CAUSE   When a breakpoint is set, a software interrupt instruction is actually placed at the address specified. This instruction causes a jump to the monitor. These instructions cannot be written into ROM.

SOLUTION   Only in-circuit emulators allow stopping normal processing at a given ROM address. Either put the code in RAM for debugging purposes or use one of the in-circuit emulators.

4.11.2   Job Flags Field Is Incorrect When Viewing A Job Token

SYMPTOM   When you view a job token with the SDB command VT, the Job Flags field is incorrect.

CAUSE   An error in the SDB.

SOLUTION   A value of 02 for Job Flags means that parameter validation is enabeled and a value of 00 means that it is disabeled. Will be fixed in a future release.

4.11.3   Using The SDB As The Default Exception Handler in iRMX II Causes GP13

SYMPTOM   When an exceptional condition occurs and execution transfers to the System Debugger, attempting to continue execution using the debugger's "GO" command causes a General Protection error.

CAUSE   The routine which calls the debugger fails to correctly handle the stack upon return.

SOLUTION   If you need to use the System Debugger as the default exception handler, answer the DEH prompt on the Nucleus ICU screen with User, which specifies a user written exception handler. Answer the NEH prompt on the Nucleus ICU screen with the name of an object file produced by assembling the following code:

```
              NAME      Exception Handler

        CODE  SEGMENT   ER PUBLIC

              PUBLIC    RQSYSEX

RQSYSEX       PROC      FAR

              INT       3          ;Invoke the SDB via the SDM

              RET       8          ;Pop parameters off the stack

RQSYSEX       ENDP

CODE          ENDS

              END
```

4.11.4 A Backslash Can Hang The System When In The Monitor

SYMPTOM A single backslash will hang the system when using the System Debug Monitor.

CAUSE Problem in the System Debug Monitor.

SOLUTION Reset the system. This problem will be fixed in a future update/release of the System Debug Monitor.

4.12 PATHNAME PROBLEMS

4.12.1 DIR Does Not List All Files

SYMPTOM Some file names are not listed when using the Directory (DIR) command.

CAUSE Files with "R?" or "r?" in their filenames are considered invisible. This is a feature of the operating system.

SOLUTION Use the "Invisible" option of the DIR command to list these files, (that is, DIR :$: I).

4.12.2 Logical Name Does Not Exist

SYMPTOM Some Human Interface commands do not function, usually generating the error "LOGICAL NAME DOES NOT EXIST".

CAUSE Some commands assume the system disk device name is :SD:.

SOLUTION Always configure systems with :SD: as the logical name for the system device.

4.12.3 DIR / Does Not List :SD:/

SYMPTOM The "DIR /" command does not list the ":SD:/" directory.

CAUSE When you change the default directory (:$:) to a different physical disk device with the ATTACHFILE command, the root directory (/) changes. The root directory (/) always refers to the root of the device being used as the default.

SOLUTION Make specific references to :SD: to access the system device root directory.

4.12.4 Error When Attempt To Access File Or Directory

SYMPTOM   Attempting file or directory access causes errors.

CAUSE   Access has not been granted to the user ID attempting to use a file or directory.

SOLUTION   Determine the access rights with the EXTENDED option on the Directory (DIR) command. The file's owner can change the access rights by invoking the PERMIT command. Also, the access rights to directories, as well as files, should be verified.

4.12.5 Unauthorized Changes To Files Or Directories

SYMPTOM   Files and directories are being changed without the user's knowledge and/or access rights to files do not seem to work.

CAUSE   There are multiple users with the same user ID.

SOLUTION   File security can be enhanced by assigning different user IDs and default directories.

4.12.6 Filenames With A "?"

SYMPTOM   Problems arise when invoking the COPY, DELETE, or RENAME command on files with a "?" embedded in their filenames, (for example, R?LOGON).

CAUSE   A "?" is seen as a wildcard, not a filename character.

SOLUTION   Enclose this portion of the pathname in quotes, for example, COPY R"?"logon. This causes the enclosed characters to be taken literally.

4.13 HUMAN INTERFACE PROBLEMS

4.13.1 System Hangs When Attempting To Access Line Printer

SYMPTOM   An attempt to access a line printer that is either physically disconnected or is not turned on, hangs the system.

CAUSE   The driver waits forever for the interrupt from the printer.

SOLUTION   Do not use the line printer when it is disconnected or powered off. The workaround is press CONTROL-C, make the printer ready and re-issue the command.

4.13.2 System Hangs When iSBX 351 Terminal Is Turned Off

SYMPTOM   If a user using an iSBX 351 serial channel disconnects or turns off the terminal, the system will hang.

CAUSE   The iSBX 351 board bombards the iRMX 286 system with interrupts when in this condition.

SOLUTION   Do not turn off the terminal. This problem will be fixed in a future update or release.

4.13.3 Error Message "delete access required"

SYMPTOM Incorrect error message (delete access required) returned by RENAME and COPY command.

CAUSE The file's directory does not have proper access.

SOLUTION Give the file's directory delete and add permission (using the PERMIT command) for the user executing the RENAME or COPY command.

4.13.4 Cannot Boot Off A Floppy Diskette

SYMPTOM A bootable diskette has no errors from the DISKVERIFY command but the diskette will not boot.

CAUSE DISKVERIFY does not verify the bootstrap loader portion of the diskette.

SOLUTION Format the disk using the BOOTSTRAP option.

4.13.5 Unpredictable System Behavior

SYMPTOM Unpredictable system operation.

CAUSE A Human Interface command is not terminated properly. Failure to correctly terminate a command causes unknown code to be executed and may cause serious loss of data.

SOLUTION Every HI command must end with either an EIOS call "RQ$EXIT$IO$JOB" or a UDI call "DQ_EXIT". The purpose of these calls is to correctly transfer control back to the iRMX II Operating System.

4.13.6 RQCSENDCORESPONSE Or RQCSENDEORESPONSE

SYMPTOM Application program does not wait for response to RQCSENDCORESPONSE or RQCSENDEORESPONSE.

CAUSE The Human Interface includes the string length of the response string in the response$max parameter. But, if the response$max is one, the Human Interface returns without a response$string.

SOLUTION The value in response$max must equal the length of the string plus one.

4.13.7 System Time Is Incorrect After A Reboot

SYMPTOM System time and date stamp are not updated.

CAUSE The BIOS does not update the access time of any file until it is modified. The Human Interface uses the last access time of the system directory (:SYSTEM:) as the default time and date.

SOLUTION Always set the time and date when you boot your system or run shutdown before resetting or powering down your system.

4.13.8 Terminal Acts Peculiar When Running RESTORE

SYMPTOM   Terminal acts peculiar when running the RESTORE command.

CAUSE   RESTORE sends a 012H (DC2) character to :CO: before printing out the message "Directory Restored".

SOLUTION   Run RESTORE from a submit file. This problem will be fixed in a future update or release.

4.13.9 Output Redirection To A Line Printer

SYMPTOM   Redirecting output to a line printer by using the Command Line Interpreter (CLI) I/O redirection arrow (>) does not work.

CAUSE   A problem with the CLI.

SOLUTION   This will be fixed in a future update or release.

4.13.10 CLI I/O Redirection In An Alias

SYMPTOM   CLI I/O redirection cannot be incorporated in an alias.

CAUSE   A design limitation of the ALIAS command.

SOLUTION   Do not use CLI I/O redirection in an alias.

4.13.11 Continuation Lines Do Not Work In CLI Commands

SYMPTOM   A E$CONTINUATION exception occurs when using continuation lines with a CLI command.

CAUSE   CLI commands do not support continuation lines.

SOLUTION   Do not use continuation lines with CLI commands.

4.13.12 DELETE Command Executed From A Submit File Acts Peculiar

SYMPTOM   Executing a DELETE command from a submit file, with the DELETE command consisting of continuation lines and more than 256 characters, it appears that the first 256 characters after the DELETE command are lost.

CAUSE   A problem with DELETE command.

SOLUTION   Do not use the DELETE command with more than 256 characters as input. This problem will be fixed in a future update or release.

4.13.13 SHUTDOWN Hangs The System

SYMPTOM   The system hangs if you have a maximum priority of 0 for your user and you run the SHUTDOWN command.

CAUSE   An error in the SHUTDOWN command.

SOLUTION   Change the user priority.

4.13.14 Problem With The Human Interface RQCGET$PARAMETER System Call

SYMPTOM   The value of the "more" parameter of the RQCGET$PARAMETER system call is always set to zero.

CAUSE   The predict list parameter of this system call is not in CAPS.

SOLUTION   The predict list should always be in CAPS.

4.13.15 Problem With The Human Interface DIR Command

SYMPTOM   If you enter "DIR * SORT", when you are three or more levels deep in the file structure, the DIR command will fail to display any information.

CAUSE   Problem with the Human Interface DIR command.

SOLUTION   Enter a CONTROL-C to get to the Human Interface. This problem will be fixed in a future Update or Release.

4.13.16 An E$SHARE May Be Encountered When Using The "TIME" Command

SYMPTOM   An E$SHARE may be encountered if the Human Interface "TIME" commmand is executed with the "GLOBAL" option.

CAUSE   The user attempted to access the global time-of-day clock while it was in use.

SOLUTION   Retry the command.

4.13.17 Problem With The Human Interface "SET" Command

SYMPTOM   AN E$SLOT message may be encountered when you execute any Human Interface command.

CAUSE   The "SET TERMINAL=VT100" command creates a segment, but does not delete it.

SOLUTION   The system must be reset. This problem will be fixed in a future update/release of the iRMX II product.

4.13.18 An E$IOMEM Exception May Be Encountered When Doing An "ATTACHDEVICE"

SYMPTOM   An E$IOMEM exception may be encountered when doing an "ATTACHDEVICE".

CAUSE   The user has not received the requested amount of memory during the "LOGON" process and therefore has insufficient memory.

SOLUTION   Ensure that the "minimum partition-size" parameter of the user attribute file (:config:user/<user-name>) is at least 512 and this value is less than the "maximum partition-size" parameter value.

4.13.19 "COPYDIR" May Hang The System

SYMPTOM   The "COPYDIR" utility may hang the system when trying to copy :SD: to a second drive.

CAUSE   Problem with the "COPYDIR" command.

SOLUTION   This problem will be fixed in a future update/release of the iRMX II product.

4.13.20 Backup Too Slow

SYMPTOM The "BACKUP" command is extremely slow when a second tape is needed to finish the "BACKUP" process.

CAUSE Problem with the "BACKUP" command.

SOLUTION This problem will be fixed in a future update/release of the iRMX II product.

4.13.21 "NOT A NAMED VOLUME" Message May Be Generated When Attaching SCS Diskettes

SYMPTOM When attaching an SCS floppy on the system 520, you will encounter the message "NOT A NAMED VOLUME".

CAUSE The 48TPI SCS diskette was not formatted with options of MS=0 and ES=41.

SOLUTION This problem will be fixed in a future update/release of the SCS product. As a workaround, you can copy the contents of the floppy on another system (i.e., system 320), format the diskette with these options (e.g., FORMAT :A: MS=0 ES=41), and re-copy the SCS files back to the formatted diskette. This way you will create a readable floppy for the system 520.

4.13.22 General Protection Exception Encountered When Setting The Aliastable Size

SYMPTOM A General Protection exception may be encountered when setting the aliastable size from the "R'?'LOGON" file.

CAUSE A problem in the Human Interface.

SOLUTION This problem will be fixed in a future update/release of the iRMX II product.

4.13.23 General Protection Exception Encountered When Using The ">" Command

SYMPTOM A General Protection exception may be encountered when redirecting output using the built-in redirection command ">".

CAUSE A problem in the Human Interface.

SOLUTION This problem will be fixed in a future update/release of the iRMX II product.

4.13.24 EIOS Initialization Error Encountered When Auto Boot Is Used In MULTIBUS II

SYMPTOM An EIOS initialization error 40H may be encountered when bootloading automatically on a MULTIBUS II system.

CAUSE A problem with the Human Interface FORMAT command.

SOLUTION This problem was fixed by Class A zap ZAP041. U.S. customers must contact TIPS. European customers must contact their local sales office.

4.13.25 "LOGOFF" Decreases User Available GDT Slots

SYMPTOM   The "LOGOFF" command will decrease the user available GDT slots each time the user logs off.

CAUSE   A problem with the Human Interface.

SOLUTION   This problem was fixed by Class B zaps ZAP102, ZAP103. U.S. customers must contact TIPS. European customers must contact their local sales office.

4.13.26 Problem With Resetting An Agent From The Human Interface

SYMPTOM   The alias "REBOOT" does not properly reset an agent from the Human Interface.

CAUSE   The problem is under investigation.

SOLUTION   As a workaround, re-enter the "REBOOT" command. This problem will be fixed in a future update/release.

4.13.27 "SET" Command Reduces The Number Of Available GDT Slots

SYMPTOM   The built-in CLI SET command reduces the number of available GDT slots.

CAUSE   Problem with the Human Interface.

SOLUTION   This problem was fixed by Class A zap ZAP016. U.S. customers must contact TIPS. European customers must contact their local sales office.

4.14   UTILITY PROBLEMS

4.14.1   LIB286 Truncates An Object File

SYMPTOM   When creating a new library, LIB286 truncates an object file and issues an error message "INVALID OBJECT FILE".

CAUSE   The disk has overflowed.

SOLUTION   Make sure sufficient disk space is available before you invoke LIB286.

4.14.2   LIB286 Truncates Updated File - Leaves Backup File Empty

SYMPTOM   When you are updating a library, LIB286 truncates the updated file and leaves the backup file empty. LIB286 issues an error message "INVALID OBJECT FILE".

CAUSE   The disk has overflowed.

SOLUTION   Make sure sufficient disk space is available before you invoke LIB286.

4.14.3   Utility Generates "SYSTEM INTERFACE ERROR"

SYMPTOM   The BLD286, BND286, MAP286, or OVL286 utilities generate a "SYSTEM INTERFACE ERROR". Control then transfers to the iSDM 286 Monitor, which in turn issues a stack protection error. In addition, all the other tasks running under the iRMX II multi-tasking system are halted.

CAUSE   The disk has overflowed.

SOLUTION   Make sure sufficient disk space is available before you invoke the utility.

4.14.4 PL/M-286 Or ASM286 Creates An Empty Object Module

SYMPTOM The PL/M-286 compiler or the ASM286 assembler appears to finish processing but creates an empty object module. No error messages appear at your terminal; however, the appropriate error messages do accumulate in the list or error print files.

CAUSE The disk has overflowed.

SOLUTION Make sure sufficient disk space is available before you invoke PL/M-286 or ASM286.

4.14.5 BND286 Truncates Input File To Length Zero

SYMPTOM The invocation sequence ":LANG:BND286 fname.lnk noload", will cause the input file "fname.lnk" to be truncated to length zero.

CAUSE The default output file created by BND286 consists of the input filename without the extension plus the extension ".lnk". So BND286 will create the output file "fname.lnk". The output filename is the same as the input filename.

SOLUTION Specify an output filename which is different from the input filename by the OBJECT control of BND286.

4.14.6 User-Written Exception Handler Does Not Receive Control

SYMPTOM A user-written exception handler does not receive control when an error occurs.

CAUSE The exception handler was compiled using the COMPACT compiler control.

SOLUTION Compile all exception handlers using the LARGE compiler control.

NOTE

All exception handlers should be in LARGE (FAR procedures) because they are called directly from the operating system.

4.14.7 Error When Invoking SUPER From AEDIT

SYMPTOM When you invoke the SUPER command from the AEDIT V2.1 editor, the exception 034H:E$CONN$NOT$OPEN is displayed.

CAUSE Error in the AEDIT V2.1 editor.

SOLUTION Be in SUPER before invoking the AEDIT V2.1 editor.

4.14.8 BND286 V3.1 Issues Error 140

SYMPTOM When using the NODEBUG control, BND286 V3.1 issues an error message "ERROR 140: TEXT LENGTH IS GREATER THAN SEGMENT LENGTH". This condition occurs with successive invocation of BND286 on the same file with a mix of DEBUG and NODEBUG controls.

CAUSE Error in BND286 V3.1.

SOLUTION Use the same control, either DEBUG or NODEBUG, in all the successive BND286 invocations.

4.14.9 BND286 V3.1 Gets General Protection Error

SYMPTOM   BND286 V3.1 gets a General Protection error when binding an object module, whose assembly language source uses the ORG directive.

CAUSE   To be determined.

SOLUTION   Do not use ORG directives in modules assembled and bound under the iRMX 286 operating system.

4.14.10 Wrong Or Missing Version Numbers In iRMX 286-Based Languages/Tools

SYMPTOM   When running the VERSION command on the iRMX 286-based languages and tools, a wrong version number or no version number is returned. The following list shows the values returned :

Name of Lang. or Tool	Value Displayed by VERSION
ASM286 V1.3	No version number found
BLD286 V3.1	X207
BND286 V3.1	X207
LIB286 V2.0	X207
MAP286 V2.1	X207
OVL286 V1.1	X207
ASM86 V2.0	No version number found
CREF86 V1.0	No version number found
LINK86 V2.7	No version number found
LOC86 V2.5	No version number found
OH86 V1.0	No version number found

CAUSE   Error in each of the languages and tools.

SOLUTION   This problem will be fixed in future releases of the languages and tools.

4.14.11 Problem With The 86-based ICU286 Utility

SYMPTOM   The 86-based ICU286 utility fails.

CAUSE   Problem with the 86-based ICU286 utility.

SOLUTION   This problem was fixed in Update 1 by class A ZAP ZXA034. For U.S. customers, contact TIPS. For European customers, contact your local sales office.

4.14.12 The UDS Utility May Abort Prematurely

SYMPTOM   The User Device Utility (UDS) utility may abort prematurely.

CAUSE   Problem in the UDS utility.

SOLUTION   This problem was fixed in Update 2 by class A ZAPs ZXA050, ZXA051, and ZXA052.

4.14.13 Incorrect Unit Data In The .TPL File

SYMPTOM   Incorrect unit information was placed in the .TPL file when using the User Device Support (UDS) utility.

CAUSE   Problem in the UDS Utility supplied with Intel's System 120 product.

SOLUTION   This problem was fixed in Update 2 by class A ZAPs ZXA050, ZXA051, and ZXA052.

4.14.14 Problem With SCS "FETCH" Command

SYMPTOM   You cannot fetch an earlier version of an SCS file for editing purposes.

CAUSE   Problem with the SCS "FETCH" command.

SOLUTION   This problem will be fixed in a future update/release of the SCS product.

4.14.15 Error 75 Encountered When Using The SCS "FETCH" Command

SYMPTOM   Error 75 is encountered when you try to "FETCH" and edit an earlier version of an SCS file.

CAUSE   Problem with the SCS "FETCH" command.

SOLUTION   This problem will be fixed in a future update/release of the SCS product.

4.14.16 Error 16 Encountered When Using The SCS Product

SYMPTOM   Error 16 is encountered when using the SCS product.

CAUSE   SCS has problems dealing with filenames of length 12 characters or more.

SOLUTION   Make sure SCS files have 12 or less characters.

4.14.17 Error 21 Encountered When Using The SCS "GENERATE" Command

SYMPTOM   Error 21 is encountered when using the SCS "GENERATE" command.

CAUSE   A general dependency may have been specified twice in a General Dependency File.

SOLUTION   Edit the General Dependency File and delete the general dependency that is specified twice.

4.14.18 Problem With The SCS "INFORM" Command

SYMPTOM   The following SCS command will not return control to the user: INFORM file-name1 > file-name2

CAUSE   The SCS product does not recognize the redirection character ">".

SOLUTION   As a workaround, use the "TO" preposition instead of the redirection character ">".

4.14.19 The SCS Key Macro "Q" Is Not Recognized

SYMPTOM   The SCS "INFORM" command does not recognize the key macro "Q".

CAUSE   Problem with the SCS "INFORM" command.

SOLUTION   This problem will be fixed in a future update or release of the SCS product.

4.14.20 General Protection Exception Encountered When Using The SCS "GENERATE" Command

SYMPTOM   A General Protection exception is encountered when using the SCS "GENERATE" command.

CAUSE   The number of general dependencies in the General Dependency File exceeded 64.

SOLUTION   This problem has been fixed. U.S. customers may contact TIPS. European customers customers may contact their local sales office.

4.14.21 Arrays Of Length 64Kbytes And Soft-Scope 286

SYMPTOM   Soft-Scope 286 cannot display the contents of an array whose size is 64Kbytes or larger.

CAUSE   Problem with Soft-Scope 286.

SOLUTION   This problem will be fixed in a future release/update of the Soft-Scope product.

4.14.22 Soft-Scope 286 And Local Variables In Recursive Functions

SYMPTOM   Soft-Scope 286 does not recognize local variables in recursive functions.

CAUSE   Problem with the Soft-Scope 286 product.

SOLUTION   This problem will be fixed in a future update/release of the Soft-Scope 286 product.

4.14.23 Cannot Use The iRMX Virtual Terminal

SYMPTOM   If a user invokes VT and logs on to a remote iRMX II.3 system, does some work, logs out, and then terminates the VT session (via "~."), he is unable to VT to that system again.

CAUSE   Problem with Update 2 of the iRMX II.3 Operating System.

SOLUTION   As a workaround, ensure that you terminate the Virtual Terminal session (via "~.") before you log off the system. This problem has been fixed by Class B zap ZAP076. U.S. customers may contact TIPS. European customers may contact their local sales office.

4.14.24 "NOT A NAMED VOLUME" Message May Be Generated When Attaching SCS Diskettes

SYMPTOM   When attaching an SCS floppy on the system 520, you will encounter the message "NOT A NAMED VOLUME".

CAUSE   The 48TPI SCS diskette was not formatted with options of MS=0 and ES=41.

SOLUTION   This problem will be fixed in a future update/release of the SCS product. As a workaround, you can copy the contents of the floppy on another system (i.e., system 320), format the diskette with these options (e.g., FORMAT :A: MS=0 ES=41), and re-copy the SCS files back to the formatted diskette. This way you will create a readable floppy for the system 520.

4.14.25 General Protection Exception Encountered When Running "RUN86"

SYMPTOM   A General Protection exception may be encountered when using the "RUN86" utility with "LIB386".

CAUSE   A problem in the UDI.

SOLUTION   This problem was fixed by Class A zap ZAP040. U.S. customers must contact TIPS. European customers must contact their local sales office.

LANGUAGES - ASM286 V1.3

5

This chapter lists known problems related to the ASM286 language, Version 1.3. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

5.1   ASM286 Assembler Gets Caught In Loop

SYMPTOM   The ASM286 assembler gets caught in loop

CAUSE   The assembler is restricted to 23 bytes per data declaration. It gives valid error messages for up to 5000 defined bytes. After 5000, the assembler hangs.

SOLUTION   Break up data declarations so that each one defines 23 bytes or less.

5.2   ASM286 Assembler Incorrectly Assumes CS Override

SYMPTOM   The ASM286 assembler incorrectly assumes a CS override.

CAUSE   If a symbol is declared EXTRN in a module and is used later without the the appropriate segment override, ASM286 incorrectly assumes a CS override.

SOLUTION   Correct the programming error. Use the appropriate segment override.

5.3   ASM286 Assembler Outputs Incorrect Segment Information

SYMPTOM   The ASM286 assembler outputs incorrect information about a segment.

CAUSE   If an EXTRN CONST symbol is placed within a segment declarations, ASM286 outputs incorrect information regarding that segment.

SOLUTION   Place call EXTRN CONST symbol declarations outside any segment declaration.

5.4   ASM286 Assembler Hangs

SYMPTOM   Occasionally, the ASM286 assembler hangs.

CAUSE   When the macro-processor runs out of memory, it attempts to do 'garbage-collection'. Under certain conditions, it does so incorrectly, causing the assembler to 'hang up'.

SOLUTION   To avoid the problem, explicitly specify 45 or larger percent of memory for the macro-processor as described in the *ASM286 Macro Assembler Operating Instructions Manual*, page 3-6.

5.5   BND286 Issues BAD OBJECT FILE Error

SYMPTOM   Although your file assembles without any errors, BND286 issues a BAD OBJECT FILE Error.

CAUSE   The assembler creates invalid TYPEDEF pointers for structures in the following situation without reporting the problem:

1. TYPE and DEBUG controls are set.

2. There are more than 256 user-defined symbols, and at least one structure is after the 256th symbol as listed in the alphabetized symbol table listing.

SOLUTION   Any of the following workarounds will correct the situation:

1. Leave off the TYPE or DEBUG control.

2. Break up the module so it contains less than 256 user-defined symbols.

3. Name structures so they are placed at the beginning of the symbol table list.

5.6 BND286 Issues BAD OBJECT FILE Message Or INTERNAL ERROR

SYMPTOM   The binder issues a BAD OBJECT FILE message if there is only one input file and that file contains an empty segment. If a file with an empty segment and one or more other files are input into the binder, it issues an INTERNAL ERROR.

CAUSE   ASM286 should not allow a label to be defined in a segment that contains no other code. The errors result during binding if that label is also declared PUBLIC.

SOLUTION   Do not allow PUBLIC labels to exist in segments without any other code.

5.7 Errors Reported In Two Situations That Should Be Warnings

SYMPTOM   ERRORs #71 and #88 should be WARNINGS instead of ERRORS.

CAUSE   ERROR #71, Source Line Cannot Exceed 255 Characters, should appear as a warning instead of an error. The only effect it has is not listing the excess characters.

```
ERROR # 88, Text
Found Beyond End Statement-Ignored, should be a warning.
```

This error results in the extra text not being assembled.

SOLUTION   In both cases, the code produced is valid. Treat the ERRORs as WARNINGs.

5.8 ASM286 Internal Error: FAILURE DURING OBJECT GENERATION, ERROR #5

SYMPTOM   The ASM286 assembler reports an internal error while assembling a program which contains greater than 50 stack segments and greater than 75 data segments.

CAUSE   Internal processing error.

SOLUTION   Within each module limit the stack segments to 50 and the data segments to 75.

5.9 BND286 Reports Segment Mismatch Error

SYMPTOM   In order to access a named FORTRAN common block from ASM, the programmer creates a segment with the name of the common block prefixed by an @ sign and a combine type of COMMON (see "ACCESSING A FORTRAN-86 COMMON BLOCK FROM OTHER LANGUAGES," November, 1986 ;*COMMENTS*.) The ASM segment should combine with the segment produced by the FORTRAN named common block. However, during binding, ERROR 121: MISMATCHED SEGMENT ATTRIBUTES is generated.

CAUSE   When the COMMON attribute is specified in a segment declaration, ASM286 produces a segment with combine type BLANKCOMMON. Since the segment generated for the FORTRAN common block has combine type COMMON, the binder generates an error when it tries to combine the segments.

SOLUTION   Change the FORTRAN named common block to an unnamed common block. Change the name of the ASM segment to "@". For an unnamed common block the FORTRAN compiler will produce a segment with combine type BLANKCOMMON and name "@". The two segments will then be compatible.

LANGUAGES - ASM86 V3.2   6

This chapter lists known problems specific to the ASM86 Language Version 3.2. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

6.1 Implied Blank Delimiters Do Not Always Work As Expected

SYMPTOM The value passed to a macro for a parameter includes blanks preceding the parameter as part of the parameter.

CAUSE The assembler passes all characters between the previous item of a macro call and the next literal delimiter as the parameter under these conditions:

1. The parameter in the macro definition is preceded by an implied delimiter; and

2. The parameter in the macro definition is followed by a literal delimiter.

For example, a macro call defined this way:

```
%*DEFINE(FOO A,B)
```

would, when called this way:

```
%FOO      X,Y
```

supply this value:

```
"      X"
```

for the value of parameter *A* instead of the intended value, "X".

SOLUTION Rewrite your macro definitions to use literal delimiters.

6.2 LINK86 Reports An Illegal FIXUP Error

SYMPTOM When an object module is linked with the BIND option, LINK86 reports an ILLEGAL FIXUP error under these conditions:

● A group override is used on a symbol (variable or label).

● The symbol is not in the group specified.

CAUSE Under these conditions, ASM86 creates a fixup for the group override which may be linked *and* located, but may not be linked with the BIND control. The BIND control has extra restrictions on fixups and does not allow this construct.

SOLUTION Either change the group name to reflect the correct group, or change the symbol name to reflect the correct symbol.

6.3 ASM86 Generates Instructions With Too Many Bytes

SYMPTOM The assembler sometimes generates a four-byte instruction when only a three-byte instruction is necessary, placing a NOP (no operation) in the last byte.

CAUSE This problem occurs when a forward-referenced location, which can be addressed as an offset only, is used as an argument to the instruction. During pass one of the assembler, space is allocated for a four-byte instruction in anticipation of having to access a location in another segment.

Example:

```
        mov   bx, offset label
          .
          .
label   db    0ah
          .
```

SOLUTION Write a CODEMACRO which will create an instruction of the proper length if you are writing a time dependent procedure that cannot afford the extra three clock cycles caused by the extra NOP.

Example:

```
R53   RECORD RF1:5,RF2:3
codemacro  mymacro  dst:Rw,src:Db
  R53 <10111B,dst>
  dw  src
endm
  .
  .
  .
mymacro bx, offset label
  .
  .
label db    0ah
  .
```

6.4   Documentation Errors

SYMPTOM   There are two errors in the *ASM86 Language Reference Manual*, (-003).

CAUSE   Two causes for this problem are:

1. On page 7-14 in the manual, under the heading "IF Function" the first example reads:

```
%IF (%GTS(0FFH,%VAR)) THEN (MOV AX, %VAR) F1
```

2. On page 7-15 in the manual, EXAMPLE 5 reads:

```
%SET (DEBUG, 1)
%IF  (DEBUG) THEN (
MOV  AX, DEBUG_FLAG
OUT  AX, 2
)
```

SOLUTION   Corrections are as follows:

1. The first example should be read as:

```
%IF (%GTS(0FFH,%VAR)) THEN (MOV AX, %VAR) FI
```

2. The second example should be read as:

```
%SET (DEBUG, 1)
%IF  (%DEBUG) THEN (
MOV  AX, DEBUG_FLAG
OUT  2, AX
) FI
```

6.5   Unexpected Results from FBSTP Instruction

SYMPTOM   The FBSTP instruction does not round precisely as documented in the *ASM86 Language Reference Manual*, (-003).

The FBSTP instruction is used to convert the contents of the 8087 stack top to a packed decimal integer and store the result in memory. The documentation states that if the value at the top of the 8087 stack is not an integer, it is rounded by adding 0.5 to the value and chopping. The actual value added is <u>close to</u> 0.5, but not exactly 0.5. For example if the top of the 8087 stack is 11.5, FBSTP stores 11 and not 12 into memory.

CAUSE   Documentation error.

SOLUTION   Round the value explicitly before taking it off the top of the 8087 stack by using the FRNDINT instruction.

6.6   Duplicated Structures Are Unexpectedly Initialized

SYMPTOM   When a structure is declared in ASM86 and that structure is duplicated with the DUP construct, the assembler generates an initialization record even if indeterminate initialization is specified. In the example below, an initialization record is generated for test_bb.

Example:

```
NAME TEST

aa_struc STRUC
     aa_name      DB   25 DUP(?)
     aa_hours     DB   ?
aa_struc ENDS

DATA SEGMENT
     test_bb aa_struc 5 DUP (<>)
DATA ENDS

END
```

CAUSE   In chapter 3 of the assembly language reference manual where indeterminate initialization is explained, there are only five cases where no initialization record is generated. These cases are of the form:

```
<type> repeatval DUP (?)
```

where <type> is DB, DW, DD, DQ, or DT. In the example above, because test_bb is a structure and is DUPed, it is not in the required form to not generate an initialization record. So, an initialization record is created.

SOLUTION   Instead of using the DUP construct to allocate an array of structures, repeat the structure definition. In the example below, test_bb is an array of 5 aa_structs. Since there is no DUP construct, and the initialization record from the structure definition is not changed, no initialization takes place. Access to the data elements declared this way is exactly the same as to data elements declared with the DUP construct.

Example:

```
NAME TEST

aa_struc STRUC
     aa_name      DB   25 DUP(?)
     aa_hours     DB   ?
aa_struc ENDS

DATA SEGMENT
     test_bb aa_struc <>
             aa_struc <>
             aa_struc <>
             aa_struc <>
             aa_struc <>
DATA ENDS
END
```

NOTE

If there are many array elements, an assembler macro may be created to ease the duplication and definition process. In the example below, a macro is used to generate the same construct.

Example:

```
DATA SEGMENT
%SET(COUNTER,4)
    test_bb aa_struc <>
%WHILE(%COUNTER GT 0)
(                    aa_struc <>
 %SET (COUNTER, %COUNTER - 1)
 )

DATA ENDS
```

6.7 Documentation Error

SYMPTOM   Macro processing language example does not work.

CAUSE   The following example is found on page 7-15 of the *ASM86 Assembly Language Reference Manual*, Order number 122386-001.

```
%SET (DEBUG, 1)
%IF (DEBUG) THEN (
MOV AX, DEBUG_FLAG
OUT AX, 2
)
```

This example is incorrect.

SOLUTION   Below is the corrected example:

```
%SET (DEBUG, 1)
%IF (%DEBUG) THEN (
MOV AX, DEBUG_FLAG
OUT 2, AX
) FI
```

The differences are:

1. A percent sign (%) has been placed in front of the identifier DEBUG in the second line. Otherwise, DEBUG will not be interpreted as having the value given to it in the preceding SET statement.

2. The operands for OUT in the fourth line have been reversed to conform to the syntax for OUT.

3. FI has been added to the end of the last line to correctly terminate the macro call.

6.8 Documentation Error

SYMPTOM   The description of the operation of the XCHG instruction is incorrect.

CAUSE   The *ASM86 Assembly Language Reference Manual*, Part No. 122386-001, page 6-105, contains a misprint in the description of the XCHG instruction. For opcode 86, the operation is shown as Mem8 <--> Mem8. For opcode 87, the operation is Mem16 <--> Mem16. This is incorrect.

SOLUTION   The operation of the instruction should be Mem8 <--> Reg8 and Reg16 <--> Mem16 respectively. The XCHG instruction is not able to handle two memory operands.

6.9   Documentation Error

SYMPTOM The description page for DAA the has wrong title.

CAUSE Page 6-38 of the *ASM86 Assembly Language Reference Manual*, Part No. 122386-001, contains a misprint. The instruction that is being described is the DAA instruction, but the title at the top of the page is CWD.

SOLUTION The title of the page should be DAA.

6.10   NOP Generated for Forward Referencing MOV Instruction

SYMPTOM When the offset of a forward referenced label is moved into the BX register, the assembler generates a NOP at the end of the instruction. Below is some code that shows the problem.

```
NAME    FORWARDREF

ASSUME CS:CODE

CODE    SEGMENT
            MOV BX, OFFSET LAB1
        LAB1:
CODE    ENDS

END
```

SOLUTION Move the label so that it is no longer forward referenced or use a register other than BX.

LANGUAGES - BLD286 V3.2 7

This chapter lists known problems specific to the BLD286 product, V3.2. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

7.1 ERROR 313 Issued During Incremental Build

SYMPTOM BLD286 issues ERROR 313: Internal Processing Error while doing Incremental Build.

CAUSE Illegal TYPE definitions output by BLD286 on the earlier Incremental Build.

SOLUTION Use the NOTYPE control.

7.2 Erroneous Value Of Limit If Item-ID Is Used

SYMPTOM BLD286 does not initialize the LIMIT field of the SEGMENT definition if the LIMITOF (GDT) construct is used.

CAUSE Internal processing error.

SOLUTION Explicitly specify the limit.

7.3 WARNING 247 And ERROR 158 Issued

SYMPTOM BLD286 issues WARNING 247 and ERROR 158, if an empty segment is input to the builder.

CAUSE BLD286 does not process empty segments.

SOLUTION Use the CREATESEG definition to define a segment.

7.4 ERROR 117 When Building With An Export File

SYMPTOM BLD286 issues ERROR 117 while building with an exported file from a previous build.

CAUSE Conflicting entries in tables.

SOLUTION Make sure that entries in tables of the two builds do not conflict.

7.5 ERROR 313 While Building With An Exported File

SYMPTOM ERROR 313 is issued while building with an exported file from an earlier build.

CAUSE The exported item is a public symbol that is also an entry point to a TSS.

SOLUTION While creating the export file, include the module name that contains the public symbol.

LANGUAGES - BND286 V3.1 **8**

This chapter lists known problems specific to the BND286 product, Version 3.1. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

8.1   First Input File Is Overwritten

SYMPTOM   The first input module in the BND286 invocation is overwritten and becomes of length zero.

CAUSE   When using the NOLOAD control, BND286 does not check to determine if the first input module will be overwritten by the .LNK output module generated. If the first input module happens to be a .LNK file, the output file BND286 will attempt to create will destroy the input module before it is read.

SOLUTION   Use the OBJECT control to force an output file of a different name than that of the input file. Be careful to use the correct syntax for the OBJECT control. If a syntax error occurs in the BND286 invocation, the output file (.LNK) may be destroyed before the syntax error is detected.

8.2   System Interface Error Is Issued

SYMPTOM   A System Interface Error is issued while executing a BND286 command. Following the error, control will pass to the iSDM™ monitor.

CAUSE   If any output file of the binder is directed to a storage device that is full, the error will be issued.

SOLUTION   Reset the system then delete some old files to provide more space on the output device.

8.3   General Protection Error Issued

SYMPTOM   A General Protection Error is issued by BND286

CAUSE   Using BND286 to bind an object module (ASM) which has an ORG statement in it causes a General Protection Error.

SOLUTION   Do not use ORG directives with iRMX® ASM286

8.4   Wrong Version Number Issued By VERSION Command

SYMPTOM   A wrong version number is issued by the VERSION command.

CAUSE   The iRMX system reads a full block (512 bytes) and the binder does not erase the unused extra bytes.

SOLUTION   Ignore the VERSION command's version number.

8.5   E$MEM Issued By The iRMX System While Binding

SYMPTOM   E$MEM is issued by the iRMX system while using BND286.

CAUSE   Improper linking of the binder.

SOLUTION   Contact TIPS for workaround.

8.6   False ERROR 125: Same Symbol Defined To Be In Different Segments

SYMPTOM   A false ERROR 125 is issued by the binder if the object files are bound in a specific order.

CAUSE   Symbol processing error.

SOLUTION   Change the order of the files in the invocation line.

LANGUAGES - BND286 V3.2 9

This chapter lists known problems specific to the BND286 language, Version 3.2. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

9.1   First Input File Is Overwritten

SYMPTOM   The first input module in the BND286 invocation is overwritten and becomes length zero.

CAUSE   When using the NOLOAD control, BND286 does not check to determine if the first input module will be overwritten by the .LNK output module generated. If the first input module happens to be a .LNK file, the output file BND286 will attempt to create will destroy the input module before it is read.

SOLUTION   Use the OBJECT control to force an output file of a different name than that of the input file. Be careful to use the correct syntax for the OBJECT control. If a syntax error occurs in the BND286 invocation, the output file (.LNK) may be destroyed before the syntax error is detected.

9.2   Wrong Version Number By VERSION Command

SYMPTOM   A wrong version number is issued by the VERSION command.

CAUSE   iRMX reads a full block (512 bytes) and the binder does not erase the unused extra bytes.

SOLUTION   Ignore the VERSION commands version number.

9.3   Interrupt 13: GENERAL PROTECTION ERROR

SYMPTOM   BND286 issues Interrupt 13: GENERAL PROTECTION ERROR.

CAUSE   Internal processing error.

SOLUTION   Use the NODEBUG switch in the invocation line.

LANGUAGES - iC-286 V3.2 **10**

This chapter lists known problems specific to the iC-286 language, Version 3.2. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

10.1   Compiler Generates Error: "lvalue required"

SYMPTOM   The compiler is unable to compile code that attempts to access function-returned structure members. It issues the following message and aborts:

```
"lvalue required"
```

Example:

```
struct list { int f1; double f2; float f3; };
main()
{
struct list r1,r3, rfun();

r3.f3 = 1.5;
r1 = r3;
if (r3.f3 == rfun (r1).f3)
printf("Pass\n")
}
```

CAUSE   Compiler restriction at this time.

SOLUTION   Use an intermediate step to assign the pointer to the structure returned by the function. Then access the structure member through the structure type variable.

Example (with reference to above example):

```
{
struct list r1,r2,r3 rfun();

r3.f3 = 1.5;
r1 = r3;
r2 = rfun(r1);

if (r3.f3 == r2.f3)
printf("passed\n")
}
```

10.2   Fatal Error, No Match, OP=41

SYMPTOM   The compiler generates the error message: "Fatal Error, No Match, OP=41" while compiling the following program.

Example:

```
struct {
        char      a:3;
        char      b:b;
        } *x;


struct {
        char      e;
        char      f;
        } *y;

proc() {
    if ( y->e != x->a )
        ;
    }
```

CAUSE   Internal compiler error occurs while comparing a normal char variable with a char bit field.

SOLUTION   Declare a char bit field with a field width equal to the size of the variable.

Example (with reference to above example):

```
struct {
        char    a:3;
        char    b:6;
        } *x;

struct {
        char    e:8;    /* declare bit field
                            the width of the variable */
        char    f;
        } *y;

proc() {
    if ( y->e != x->a )
        ;
    }
```

10.3 In cc2b: Fatal error:missing blkend

SYMPTOM The compiler issues a fatal error while compiling a file which contains a function. The function's return value is a function call to a procedure previously defined in the file and also defined within the function.

For example:

```
int proc1(a)
int a:
    {
    return (a+2);
    }

int proc2(b)
int b:
    {
    int x,proc1();
    return(b + proc1(x));
    }
```

CAUSE Internal compiler error in the optimization phase of the compiler.

SOLUTION Remove the declaration from within the function or move the called function to after the calling function.

For example:

```
int proc2(b)
int b:
    {
    int x,proc1();
    return(b + proc1(x));
    }

int proc1(a)
int a:
    {
    return (a+2);
    }
```

10.4   Line Numbers In Debuggers Do Not Correspond With Source Listing

SYMPTOM   While debugging iC286 object modules under the control of a symbolic debugger, line numbers in the source listing may not correspond with the line numbers in the debugger. The inconsistency occurs on modules compiled with DEBUG and OPTIMIZE(0) controls.

CAUSE   The iC286 compiler changes the order of the instructions during the optimization process.

SOLUTION   Use ASM286 control to obtain an assembly language listing for use in debugging your object modules.

10.5   String Size Overrides Array Limit

SYMPTOM   The compiler allows an explicitly-dimensioned char array that is declared external to main to be initialized with a string larger than the array dimension. The size of the resulting array is the number of characters in the initializer string + 1 (for the null string terminator). For example, the compiler accepts the following declaration.

```
char foo[10]={"AbCDEFGHIJKLMNOPQR"};
```

```
main()
{   }
```

The length of foo is 19, overriding the explicit dimension 10.

CAUSE   Internal compiler error.

SOLUTION   Initialize external char arrays carefully, ensuring that the initializing string and null string terminator do not exceed the size of the array.

10.6   iC286 Compiler Generates Bad Code

SYMPTOM   The iC286 compiler generates incorrect opcodes while compiling the following program. The opcode following the call to plmfunct specifies SUB SP,SP rather than SUB AH,AH.

```
extern alien unsigned char plmfunc();

main()
{
  unsigned int a;

  while (plmfunct() != 0 )
        a= 0xa5a5;

}
```

CAUSE   Internal compiler error while generating opcodes.

SOLUTION   Remove the "unsigned" from the declaration of the function.

10.7 FATAL ERROR NO MATCH, OP=61

SYMPTOM The compiler issues a FATAL ERROR, NO MATCH, op=61 message while compiling the following program:

```
main()
{
  struct { char x; char y; } *i;
  struct { char x; char y; } *j;

  unsigned int a;

  a += i-j;

}
```

CAUSE Internal compiler error while performing pointer subtraction.

SOLUTION Remove the "unsigned" from the declaration of the variable a.

10.8 The 287 Chip Is Not Initialized On Program Startup

SYMPTOM The LQMAIN.AS2 file provided with the iRMX 286 iC286 compiler does not include a "CALL INIT87" instruction to initialize the 287 chip.

CAUSE The call was omitted from the file.

SOLUTION Edit the LQMAIN.AS2 file. Include a "CALL INIT87" in the file prior to the standard I/O initialization (_fsetup()). Assemble LQMAIN.AS2 and include the resulting LQMAIN.OBJ in your bind.

Example:

```
          This is the file LQMAIN.AS2

               .
               .
               .
     mov   word ptr argv[bx],0      ; end of the
     mov   word ptr argv[bx+2],0    ; argv array.

     call init87                    ; insert call to init87 here

; _fsetup();
     sub   bp,bp                    ; Clear BP so setjmp works.
     call  _fsetup                  ; Set up standard I/O.
               .
               .
               .
```

10.9 Memory Is Corrupted

SYMPTOM   Memory is corrupted after floating point store instructions are executed.

CAUSE   The iC86 and iC286 compilers do not insert an FWAIT instruction prior to the epilog of a function which contains a floating point store instruction. The following scenario can occur:

The last instruction in a function is a floating point store, FSTPF, which can take 80 clocks to complete (on a 286/10 this instruction might take 20 microseconds). The next instruction is the stack cleanup, "MOV SP,BP". Without an intervening FWAIT instruction, this MOV instruction is permitted to execute. The SP now holds the value it had prior to the CALL to this function. When the floating point store instruction completes later (asynchronously), it might then store its result to an incorrect memory location that is now being used by another function.

SOLUTION   This problem is under investigation. Customers should call the TIPS hotline for more information. European customers should contact their local sales office.

10.10 No Error Message Issued When Referencing an Undefined Structure Member

SYMPTOM   The following example compiles without error. The example references a member of structure 'y' which is not defined. The error 'member "b" is not defined' should be reported by the compiler.

Example:

```
main()
{
struct {int a; int b;} x;
struct {int c; int d;} y;

x.a=5;
y.b=9;

}
```

CAUSE   Internal error in the compiler.

SOLUTION   Be sure to reference only structure members which have been defined.

10.11 CC286 Generates General Protection Error

SYMPTOM   Getting a General Protection Error when compiling the following program.

Example:

```
char str[sizeof("1234")];

            main()
            {
              int i;
              i = 0;
            }
```

CAUSE   The use of the sizeof operator in the first statement of the program.

SOLUTION   Define a dummy variable before the statement or replace sizeof with the actual number.

10.12   CC286 Generates General Protection Error With DEBUG Control

SYMPTOM   General Protection error when compiled with DEBUG control.

```
union { char i[8192];
        int a[4096]; } *problem;

main()
{
}
```

CAUSE   The General Protection error occurs when the union size is greater than or equal to 8192 and a pointer to that union is declared.

SOLUTION   Compile with no debug or decrease union size. Other solutions: contact Intel.

10.13   Bad Object File Produced By CC-286 Compiler

SYMPTOM   Bad object file is produced by the CC-286 compiler when a very complex storage definition is used.

CAUSE   When compiling a program that has a very complex storage definition a internal limit is exceeded, producing the bad object file. This only occurs when the DEBUG option is specified.

SOLUTION   Reduce the complexity of the complex storage definition or compile without the DEBUG option. Contact Intel Software Support for other possible solutions.

10.14   Compiler Crashes on Host Environment

SYMPTOM   Compiler crashes on the host environment.

CAUSE   Under investigation.

Example:

```
typedef union
{
    struct
    {
        unsigned     dest_sse_id : 2;
    } link_adr_bits;

    unsigned char link_adr_byte;
}
link_adr_type;

static void lc_initialize_pl(lnk_adr)
link_adr_type    *lnk_adr;
{

    lnk_adr->link_adr_bitsst_sse_id = 0; } .de
```

SOLUTION   Under investigation. Contact Intel Software Support for possible solutions.

10.15 General Protection Error While Compiling

SYMPTOM General Protection error is issued while compiling.

CAUSE Using the "sizeof" operator as part of the first statement in the file.

Example:

```
static char str[sizeof("1234")];
main()
{
    int i;
    i=0;
}
```

SOLUTION Don't use the "sizeof" operator in the first statement in the file.

CAUSE When compiled with DEBUG and a typedef of a union is defined with an array of size 8192 or more.

Example:

```
typedef union {char i[8192];
                int  a[4096];}u;
u *ptr;
```

SOLUTION Under investigation. Call Intel Software Support for possible solutions.

LANGUAGES - iC-286 V4.1   11

This chapter lists known problems specific to the iC-286 language, Version 4.1. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

11.1 Structure Causing Type Mismatch

SYMPTOM BND286 reports a type mismatch error when a C program is linked with a file written in another language and both files reference the same structure.

CAUSE By default, the ALIGN control is turned on by the C compiler. This causes extra bytes to be added to C structures.

SOLUTION Use the NOALIGN compiler control.

11.2 Invalid Dictionary Access, Case 3

SYMPTOM An "invalid dictionary access, case 3" is reported by the compiler before the host system halts.

CAUSE This error occured when a typedef was used to define a structure that contained, as one of its elements, another structure that has not been defined yet.

SOLUTION Do not forward reference structure definitions. Make sure structures are already defined before using them in later definitions.

11.3 Statement Has No Apparent Effect

SYMPTOM The diagnostic remark "statement has no apparent effect" is incorrectly reported.

CAUSE This occurs when using the putc() and putchar() functions while the compiler control DIAGNOSTIC(0) is being used.

SOLUTION Ignore the remark. Proper code is still generated.

11.4 Data Segment Size Of 0FFFFH Is Created

SYMPTOM No error is reported, yet an incorrect data area size of 0FFFFH is reported at the end of a list file.

CAUSE Storage is being reserved for an undefined structure. For example, the following code compiles without error even though struct_type is undefined:

```
struct struct_type s1;
main()
{
}
```

When the list file is examined, the following information is given:

```
DATA AREA SIZE          = FFFFH   65535D
```

SOLUTION Be sure to define a structure before reserving storage for it in the data segment.

11.5   User Program Executes Function Out Of Sequence

SYMPTOM   A program compiles and binds without error but does not execute properly. A function may be called at the wrong time.

CAUSE   If a local static function has been prototyped, make sure the function definition exists. The following file demostrates this problem:

```
#include <stdio.h>

static int proto(int);  /* prototype declared  */
                        /* but function missing */
hello_func()
{
printf("hello\n");
}

main()
{
proto(1);               /* missing function called */
}
```

This file compiles without error but when it is executed, the hello_func() gets executed instead, since it is the first function to follow the prototype. The compiler should have reported that the proto() function was missing.

SOLUTION   Make sure local functions are defined when corresponding prototypes are also used.

11.6   Cannot Initialize

SYMPTOM   "Cannot initialize" is reported while trying to initialize one structure with the address of another structure.

CAUSE   The following example generates this error:

```
struct { char buf1[10]; } S1;
struct { char *buf2; }   S2 = { S1.buf1 };
```

SOLUTION   This problem will be fixed in the next release of the compiler.

11.7   System Hangs While Compiling

SYMPTOM   The host system hangs while compiling.

CAUSE   If a syntax error exists near the beginning of a source file, the compiler can hang the system. The following code is an example of this problem:

```
extern test()       /* missing ';' */
#include <stdio.h>
main()
{
 test();
}
```

SOLUTION   Check for syntax errors near the beginning of the source file if the host system hangs while compiling.

11.8 S_IREAD and S_IWRITE Not In io.h

SYMPTOM The macros S_IREAD and S_IWRITE should be in the header file io.h. The chmod() library function uses these two macros when it is executed on an iRMX Operating System.

CAUSE These two macros are defined in another header file.

SOLUTION Look for these two macros in the sys/stat.h file.

11.9 Hex Numbers Not Printed Properly

SYMPTOM Hex numbers may not be printed properly by the printf() function.

CAUSE This problem only occurs when the uppermost bit of the hex number is a 1.

SOLUTION Typecast the hex number to a larger variable type inside the printf() statement.

11.10 Sscanf() Not Starting Correctly

SYMPTOM The sscanf() function waits before doing its conversion.

CAUSE A carriage return is needed from stdin before sscanf() will do its conversion.

SOLUTION This problem can be worked around by using freopen() to redirect stdin before the sscanf(). For example:

```
freopen(":BB:","r",stdin);
```

11.11 The ctime() Function Is Not Working Properly

SYMPTOM Parts of the hours:minutes:seconds portion of the string returned by ctime() may be missing.

CAUSE If the hours, minutes, or seconds happen to equal "00", the zeros are removed from the string. For example, if the time equals 12:00:34, then 12::34 is returned.

SOLUTION This problem will be fixed in the next version of the iC286 libraries.

11.12 GP Error Occurs While Doing String Comparisons

SYMPTOM A GP error can occur while executing iC286 code that compares two similar strings of unequal length.

CAUSE The following code can cause this error:

```
char *s1 = {"abc"};
char *s2 = {"abcd"};

strcmp(s1,s2);
```

SOLUTION This problem only occurs when the first string is shorter than the second string.

11.13 Out Of Memory Or Out Of Slots Error Occurs

SYMPTOM An out of memory or out of descriptor table slots error is reported at run time. On the iRMX II operating system, these errors would be reported as E$MEM and E$SLOT.

CAUSE The sscanf() and sprintf() functions create semaphores each time they are called. These semaphores never get deleted. If either of these two functions are called many times, the host system eventually runs out of memory or slots.

SOLUTION The next release of the iC286 libraries will delete these semaphores.

11.14 RQ$CREATE$IO$JOB Prototyped Wrong

SYMPTOM RQ$CREATEIOJOB is prototyped wrong in rmxc.h header file.

CAUSE The poolmin and poolmax parameters should be protoyped as WORD variables not DWORD's.

SOLUTION Change the prototype. This problem is fixed in V4.2.

11.15 Type Mismatches For iRMX System Calls

SYMPTOM BND286 reports type mismatch errors for RMX system calls that are passed or return a selector.

CAUSE The CRMX2 libraries use the integer data type rather than the selector data type.

SOLUTION These type mismatches can be ignored. The selector data type will be used in the next release of the CRMX2 libraries.

11.16 Unable To Use sizeof() In typedef

SYMPTOM The compiler reports errors when using the sizeof() function to get the size of a variable type inside of a typedef.

CAUSE The following code is not allowed by the compiler:

```
typedef char array_type[sizeof(long)];
array_type array4;
```

SOLUTION Pass a variable rather than a variable type to the sizeof() function. The following code is accepted by the compiler:

```
long byte4;
typedef char array_type[sizeof(byte4)];
array_type array4;
```

11.17 time() And clock() Not Working Under iRMX

SYMPTOM The time() and clock() functions are called and no errors are reported when linked, but invalid results are returned.

CAUSE None of the functions listed in the header file time.h are supported under the iRMX Operating System, although some dummy routines can be found in the CRMX libraries by these names.

SOLUTION Do not use these functions under the iRMX Operating System. This problem is fixed in V4.2.

11.18   Error Including rmxerr.h And errno.h At The Same Time

SYMPTOM   When the files rmxerr.h and errno.h are included at the same time, an "illegal macro definition" error is reported.

CAUSE   Both header files define the macro EEXIST, but with different values.

SOLUTION   Comment out the macro definition that is not needed by the application.

11.19   E$MEM Error Reported When Compiler Is Invoked

SYMPTOM   The release notes state that a minimum of only 374K bytes of memory are needed. Yet, the minimum has to be increased to over 700k in order to run the compiler.

CAUSE   The application loader for RMX II Release 3 cannot properly load the overlays used by the iC286 compiler. This results in Release 3 needing a minimum of more than 700k in the user partition.

SOLUTION   Zaps are available that fix the application loader for RMX II Release 3. These zaps can be obtained from the people who support RMX II. The application loader for Release 4 of RMX II does not have this problem.

11.20   GP Error Compiling A Program With Indirect Function Calls

SYMPTOM   A GP error is reported while compiling a program that makes an indirect call to a function that returns a selector.

CAUSE   The following code causes this problem:

```
#include <i86.h>

selector sel;
selector (*p_func)();
selector (**pp_func[])() = (&p_func);

main()
{
sel = (**pp_func[0])();
}
```

SOLUTION   Avoid making indirect calls to functions returning a selector. This limitation will be fixed in the next release of the compiler.

LANGUAGES - CREF86 V1.0 **12**

This chapter lists known problems specific to the CREF86 product, Version 1.0. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

12.1   Title String In Output Contains Corrupted Data

SYMPTOM   Corrupted or meaningless data is contained in the TITLE string of CREF86 output listing.

CAUSE   If the title in the CREF86 output contains exactly the same number of characters as the page width, CREF86 sometimes dumps meaningless data in the output listing.

SOLUTION   To avoid the problem, keep the length of the title less than the page width.

12.2   Lines In Output Listing Contain Corrupted Data

SYMPTOM   Corrupted or meaningless data is contained in some lines of the CREF86 output listing.

CAUSE   If a line in the CREF86 output contains exactly the same number of characters as the page width, CREF86 sometimes dumps corrupted data in the output listing.

SOLUTION   To avoid the problem, keep the length of the line less than the page width.

12.3   CREF86 INTERNAL ERROR Message Issued

SYMPTOM   The CREF86 INTERNAL ERROR message is issued when the total number of TYPDEFs in the input modules exceeds the 1024 limit.

CAUSE   A CREF86 internal limitation exists which prevents more than 1024 TYPDEFs (structure definitions, etc.) from being handled simultaneously.

SOLUTION   Submit fewer modules to a single CREF86 invocation to reduce the total number of TYPDEFs it must handle or reduce the total number of TYPDEFs in your procedure(s).

LANGUAGES - FORTRAN 286 V3.1 **13**

This chapter lists known problems specific to the FORTRAN 286 product, Version 3.1. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

13.1 Fatal Error Message "LABEL ADDRESS AND CURRENT LOCATION COUNTER DISAGREE IN PHASE P"

SYMPTOM Fatal error message "LABEL ADDRESS AND CURRENT LOCATION COUNTER DISAGREE IN PHASE P" in a DO loop when using OPTIMIZE level 2 or 3; GENERAL PROTECTION exception when using OPTIMIZE level 2 or 3; Incorrect code generated when using OPTIMIZE level 2 or 3.

CAUSE A compiler error can occur while optimizing a DO loop under certain, as yet, unidentified conditions.

SOLUTION Under investigation. Place a statement label on the first statement inside the DO loop and end the loop using a labeled CONTINUE statement. If restructuring the DO loop does not eliminate the error(s), compile with OPTIMIZE(1) instead. Will be fixed in next release of Fortran86/286.

13.2 Error Message "NEAR /.._DATAn/, ERROR: NO MESSAGE FOR CONDITION C-06CH" During Compilation

SYMPTOM The error message "NEAR /.._DATAn/, ERROR: NO MESSAGE FOR CONDITION C-06CH" during compilation of a program containing many large local arrays.

CAUSE A compiler error causes duplicate segment names to be generated when over 11 local data segments are needed for a single module

SOLUTION Place some of the local arrays in separate, named COMMON blocks, such that no more than 11 local DATA segments are allocated (not counting overflow segments resulting from arrays > 64K). Use the "Storage Requirements" list near the end of the .LST file as a guide. Will be fixed in next release of Fortran86/286.

13.3 Fatal Error Message "OBJECT MODULE GENERATOR ERROR IN PHASE P"

SYMPTOM The fatal error message "OBJECT MODULE GENERATOR ERROR IN PHASE P" occurs for a module with a very large number of segments.

CAUSE The total number of segments generated in the object module exceeded the capacity of an internal buffer. Both the STORAGE and DEBUG controls can cause additional segments to be generated.

SOLUTION Reduce the total number of segments generated for the module by reducing the sizes of arrays or splitting up the module into multiple modules. Will be fixed in next release of Fortran86/286.

13.4 Modules Linked With RTN286.LIB Or RTNULL.LIB Result In PROTECTION EXCEPTION Or Destroyed Data

SYMPTOM PROTECTION EXCEPTION results or data are destroyed when executing modules linked with RTN286.LIB or RTNULL.LIB .

CAUSE The run-time initialization procedure needlessly initializes its DATA segment, which is defined too small.

SOLUTION In the final link/build step, increase the segment size using the following control:

```
SEGSIZE(TQNULLDATAAREA(12H))
```

This problem will be fixed in next release of Fortran86/286.

13.5 Object Program Hangs System or Aborts With PROTECTION EXCEPTION

SYMPTOM The object program hangs up the system or aborts with a PROTECTION EXCEPTION.

CAUSE The compiler always generates a far call for a recursive function call in a COMPACT procedure.

SOLUTION Place the function name in the EXPORTS list of the COMPACT subsystem definition and recompile. Will be fixed in next release of Fortran86/286.

13.6 Microsoft C Calls Result in PROTECTION EXCEPTION Errors or Destroy Data

SYMPTOM Microsoft C calls result in PROTECTION EXCEPTION errors or destroy data when using the RAM (or CONST IN DATA) option.

CAUSE The compiler erroneously initializes the DS register as if constants were always in the CODE segment.

SOLUTION Use the ROM option for procedures making calls to Microsoft C procedures. Will be fixed in next release of Fortran86/286.

13.7 Warning Message "F183 PARAMETER TYPE MISMATCH WITH EARLIER INVOCATION OF PROCEDURE" or "TYPE MISMATCH"

SYMPTOM Compilation warning message "F183 PARAMETER TYPE MISMATCH WITH EARLIER INVOCATION OF PROCEDURE" or link/build warning "TYPE MISMATCH".

CAUSE The size of INTEGER and LOGICAL actual arguments is defined to be the size of the expression result or the default size (see STORAGE compiler control), whichever is larger. This rule sometimes results in the above warnings. The current compilers apply and enforce this restriction more consistently than did previous releases.

SOLUTION These warnings may be ignored if the size of each INTEGER or LOGICAL dummy argument of the called subprogram is declared to be smaller or the same as that of the corresponding actual argument in every reference to that subprogram. They can be eliminated by assigning the expression to a variable of the correct size for use as the actual argument.

13.8 Syntax Error During Assembly Of The ASM286 Code

SYMPTOM When developing code to share a named COMMON block between a Fortran286 module and an ASM286 module per the Fortran86/186/286 documention, assembly of the ASM286 code results in syntax errors.

CAUSE The Fortran documentation is in error in regard to ASM286 segment declaration conventions - classnames do not exist.

Where "FTNCOMMON" represents a valid Fortran named COMMON block, this segment definition is illegal under ASM286:

```
@SHRDATA SEGMENT COMMON '@FTNCOMMON'
```

SOLUTION Data may be shared between Fortran286 and ASM286 only via an unnamed COMMON block, i.e. (in the ASM286 module):

```
@ SEGMENT RW COMMON
```

13.9 Fortran CLOSE Statement Aborts Execution, Produces Error Message "8002H E$TYPE"

SYMPTOM At run-time, a Fortran CLOSE statement aborts execution and produces the RMX II error message "8002H E$TYPE".

CAUSE A compiler error results in an attempt to read a selector containing invalid information for a temporary buffer.

SOLUTION There is no solution at this time. Will be fixed in next release of Fortran86/286

13.10 Successive CLOSE/OPEN File Statements Abort Execution, Produce Error Message "000CH E$SLOTS"

SYMPTOM At run-time, successive CLOSE/OPEN file statements abort execution and produce the RMX II error message "000CH E$SLOTS".

CAUSE A compiler error results with successive CLOSE/OPEN statements, exhausting available GDT slots.

SOLUTION There is no solution at this time. Avoid unnecessary CLOSE/ OPEN statements; use REWIND where possible. Will be fixed in next release of Fortran86/286.

13.11 Problem When Using "optimize(1)" At Compile Within Subroutine

SYMPTOM This problem only occurs when using "optimize(1)" at compile, and only occurs within a subroutine: When integer array elements are initialized within an if_else_endif construct, and the value assigned is composed of the array element plus an integer value, erroneous code is generated. This consists of an extraneous "push es" in the code generated for the statement following the "else", and an extraneous "pop cx" generated for the "return" statement. The following code fragment reproduces the problem:

```
PROGRAM MAIN
INTEGER*4 FOO(2000)
CALL BOO(0.0,1.0,FOO)
END
SUBROUTINE BOO (DX,DY,FOO)
INTEGER*4 FOO(2000)
   I=1
   IF (DX .GT. DY) THEN
     FOO(I) = FOO(I) + 1
   ELSE
       FOO(I) = FOO(I) - 1
   ENDIF
RETURN
END
```

CAUSE Under investigation at this time.

SOLUTION Place a statement label preceding the "ELSE", ie:

```
10  ELSE
```

Placing a statement label at the "else" eliminates the erroneous code. Or compile using optimize(0).

LANGUAGES - LIB286 V2.1 14

This chapter lists known problems specific to the LIB286 product, Version 2.1. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

14.1 Input File Corrupted And Backup File Is Length Zero

SYMPTOM A library file being updated is corrupted and the backup file is of length zero.

CAUSE This problem occurs if the library file being updated is read from and sent to a full storage device.

14.2 Librarian Enters iSDM™ Monitor

SYMPTOM While using the librarian, control transfers to the iSDM monitor.

CAUSE This will occur if the module or publics listings are sent to a file on a full storage device.

14.3 Internal Processing Error Issued

SYMPTOM An internal processing error is issued.

CAUSE This error is issued if a user does not have read access rights on the file being input into the librarian.

SOLUTION Give the file read access rights.

LANGUAGES - LIB86 V2.1 15

This chapter lists known problems specific to the LIB86 product, Version 2.1. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

15.1 INVALID CHECKSUM Error Issued

SYMPTOM   An INVALID CHECKSUM error message is issued.

CAUSE   When the internal representation of a number of PUBLIC symbols in the object file creates a boundary line condition (exactly 1K for symbols in one segment), the error is erroneously issued.

SOLUTION   To avoid this problem, either add a public symbol or remove one by using PUBLICS EXCEPT (public_symbol) at link time.

LANGUAGES - LINK86 V2.7 **16**

This chapter lists known problems specific to the LINK86 product, Version 2.7. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

16.1   BAD OBJECT FILE Created

SYMPTOM   A BAD OBJECT FILE is created.

CAUSE   If the FASTLOAD control is used at link time, and there are approximately 350 or more segments, the linker produces a bad object file.

SOLUTION   Avoid using the FASTLOAD control.

16.2   No LINK86 Errors But Incorrect Results At Run Time

SYMPTOM   The linker issues no errors, but your program does not give the expected results.

CAUSE   If a variable is located outside its associated data segment, the LINK86 product may not catch the error.

SOLUTION   Make sure all variables are in the correct data segment.

16.3   No LINK86 Errors - Problems At Run Time

SYMPTOM   The linker does not issue any errors but you have problems at run time.

CAUSE   If you reduce segment size (via the SEGSIZE control) so much that some initialized data resides beyond a segment limit, the linker may not catch the error.

SOLUTION   Make sure the SEGSIZE control does not reduce a segment size.

16.4   LINK86 Issues Error 26 "NOT AN OBJECT FILE"

SYMPTOM   The linker issues an Error 26 "NOT AN OBJECT FILE".

CAUSE   If a nonexistent module is specified as being contained in a library in the linker's invocation, LINK86 issues an Error 26 for the next module in the input list stating that it is "NOT AN OBJECT FILE".

SOLUTION   Avoid linking in a nonexistent module from a library.

16.5   FORTRAN DATA Statements Occasionally Do Not Work

SYMPTOM   Successive FORTRAN array elements do not initialize properly with DATA statements when the values to which they are being initialized are the same.

CAUSE   If the FASTLOAD control is used at link time, the data initialization for successive data elements initialized to the same values could be subsequently overwritten with zeroes. Occurrence is governed by the optimal method of initialization chosen by the FORTRAN compiler.

SOLUTION   Avoid using the FASTLOAD control.

16.6   Error 23 Issued When Linking In Large Library Of FORTRAN Modules

SYMPTOM   Error 23, BAD OBJECT MODULE is issued when linking in a large library of FORTRAN modules.

CAUSE   When trying to link a library containing more than approximately 290 FORTRAN modules, the LINK86 product will generate Error 23 for the next module in the library and abort processing.

SOLUTION   Split the library of FORTRAN modules into two or more libraries.

16.7 Error 15 Issued With Pascal-86 V3.1 Object Files

SYMPTOM   Error 15 is issued with Pascal-86 V3.1 object files.

CAUSE   Occasionally, if the number of characters which comprise all or part of the symbols of a public record meets or exceeds a boundary condition, Error 15 (Contact your Intel representative immediately) is issued. This problem can occur with large records that have more than 63 symbols of 10 characters each, or small records comprised of very long symbol names. Any other occurrence is difficult to predict because it only occurs when a buffer overflows exactly at the start of a new symbol name.

This is a known problem with the Pascal-86 V3.1 language. The compiler generates an incorrect TYPDEF code in the object module.

SOLUTION   Use the NOTYPE control while compiling or linking.

16.8 Error 23 Issued With ASM86 Object Modules

SYMPTOM   Error 23 is issued with ASM86 object modules.

CAUSE   The linker occasionally issues an invalid Bad Object File message while processing an ASM86 object module that contains bad TYPE information.

SOLUTION   Use the NOTYPE control while linking to cause the linker to ignore the bad TYPE information.

16.9 Warning 17 Issued For STACK Segment

SYMPTOM   Warning 17 is issued for STACK segment.

CAUSE   The linker will issue a Warning 17 (SEGMENT OVERFLOW) when attempting to link modules whose combined STACK requirements exceed 64K bytes.

The linker SEGSIZE control cannot be used as a workaround, since the SEGSIZE control is invalid without the simultaneous use of BIND. When used simultaneously with BIND, under these circumstances, to force a fixed stack segment size (as in SEGSIZE(STACK(48000))), Error 46 (ILLEGAL USE OF SEGSIZE CONTROL) is generated and processing is aborted. Trying to use the linker SEGSIZE control to reduce the size of the stack segment (as in SEGSIZE(STACK(-25000))) results in the appropriate Warning 42 (DECREASING SIZE OF SEGMENT), but at completion of the link process, the map file is incomplete and the object module is invalid.

SOLUTION   Some Intel language products contain a feature whereby the STACK segment can be forced to wrap around the 64K-byte boundary, thereby requesting less STACK than actually necessary. This is true of the PL/M-86 language and is also true of Pascal-86 "MAIN" modules. It is not true of Pascal-86 procedures.

To take advantage of this feature,

1. The user must declare a dummy array within each of the modules that needs to have its STACK reduced.

2. The dummy array must be declared as REENTRANT so that it is placed on the stack.

3. The dummy array must be of a sufficient size such that when its size is added to the existing stack requirements of the module, the total STACK required by the module exceeds 64K bytes.

   The compiler will wrap around and call for substantially less STACK than is actually required. The linker SEGSIZE control can then be used to adjust the STACK to the proper size.

16.10   Warning 9 NOT Issued With ASM86 Object Modules

SYMPTOM   Warning 9 (TYPE MISMATCH) is NOT issued with ASM86 object modules.

CAUSE   The linker will NOT issue a Warning 9 (TYPE MISMATCH) when processing an ASM86 object module that contains references to externally defined symbols. The assembler places a NULL-TYPE definition in the .OBJ file for an ASM86 symbol which has been declared as EXTERNAL in the ASM86 source module. NULL-TYPE will match any other type.

SOLUTION   Manually check your code when linking with ASM86 object files to ensure that type mismatches will not create any problems.

16.11   Linker Produces Larger-Than-Expected Overlaid Executables For iRMX® 86 Operating Systems

SYMPTOM   The linker produces larger-than-expected overlaid executables for iRMX 86 operating systems.

An overlaid application developed for execution on an iRMX 86 operating system may consume more memory than expected.

CAUSE   Two possible causes are:

1. You may have neglected to account for the additional overhead of the overlay software (approximately 10K bytes) in your expectations of memory consumption.

2. You have specified a larger-than-necessary value for the MAXSIZE argument of the MEMPOOL control in the final LINK86 invocation which created the executable module. The iRMX 86 loader loads overlaid modules differently. It pays special attention to the MEMPOOL(,MAXSIZE) record in the executable module and immediately requests the MAXSIZE amount of memory from the FSM upon loading. An overlaid application which has been linked with V2.7 of the LINK86 product contains all the correct information in the executable object module.

SOLUTION   When linking overlaid applications, remember to account for the software overhead of the overlay handling software and be cautious about specifying excessive amounts of memory in the MEMPOOL(,MAXSIZE) argument.

16.12   Missing Comma In Invocation Line Causes Omission Of Input Module

SYMPTOM   A missing comma in the invocation line causes omission of the input module.

CAUSE   In the following example:

```
RUN LINK86              &
:f1:LIBRARY(LIB_MOD_1)  &
:f1:MOD_1.OBJ,          &
:f1:MOD_2.OBJ,          &
:f1:LIBRARY(LIB_MOD_2)  &
to  :f1:MOD_1.LNK
```

The file :f1:MOD_1.OBJ will be omitted from the file :f1:MOD_1.LNK without any warning from the linker. This problem occurs when the comma is missing following the library reference and another reference is made to the same library later in the invocation line. The LINK86 product does not catch and properly report the syntax error.

SOLUTION   Correct the syntax of the LINK86 invocation.

LANGUAGES - LOC86 V2.5   17

This chapter lists known problems related to the LOC86 product, Version 2.5. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

17.1   Invalid Warnings At LOC86-time

SYMPTOM   You receive invalid warnings at LOC86-time.

CAUSE   A constant segment of size 0FFFEH, plus a data segment of one byte in DGROUP cause erroneous warnings.

SOLUTION   Avoid having constant and data segments with these sizes.

17.2   PURGE Control Does Not Remove All Information

SYMPTOM   Although the PURGE control is used at LOC86-time, all type information is not removed. TYPDEFs are removed, but index values in symbol definitions remain unchanged.

CAUSE   The problem only occurs if there are unresolved symbols in the LOC86 output.

SOLUTION   Specify each control explicitly (NOLINES, NOSYMBOLS, NOCOMMENTS, NOPUBLICS).

17.3   No Warning Is Given When Segments, Within a Class, Wrap Around To Low Memory Addresses

SYMPTOM   Segment base values, within a class, have both high and low memory addresses.

CAUSE   When a class would extend beyond the 1MB address range (i.e., CLASS BASE + CLASS SIZE > 1MB), only those segments which fit completely below the 1MB limit are put into high memory. The remaining segments are assigned addresses starting from 200H (see example).

EXAMPLE

```
DOS 8086 LOCATER, V2.5

CONTROLS SPECIFIED IN INVOCATION COMMAND:
ADDRESSES(CLASSES(RAM(0F8000H))) NOINITCODE

SEGMENT MAP

START    STOP     LENGTH    ALIGN    NAME     CLASS
F8000H   FAB65H   2B66H     G        DATA     RAM
FAB70H   FF98FH   4E20H     G        STACK    RAM
00200H   05BCEH   5BCFH     G        CODE     RAM
```

SOLUTION   Examine the map file from LOC86 and, if desired, reassign the class at a lower memory address.

LANGUAGES - MAP286 V3.2 18

This chapter lists known problems specific to the MAP286 product, Version 3.2. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

18.1   ERROR 114: INVALID OBJECT FILE

SYMPTOM   MAP286 issues ERROR 114: Invalid object file.

CAUSE   Incompatible Fortran Version 3.0 is used.

SOLUTION   None.

18.2   MAP286 Runs For Several Hours

SYMPTOM   MAP286 runs for several hours.

CAUSE   Insufficient memory allocated by OS.

SOLUTION   Increase memory in /CONFIG/TERMINALS file.

18.3   ERROR 302: UNKNOWN ERROR

SYMPTOM   MAP286 issues ERROR 302: unknown error while processing a loadable file.

CAUSE   Internal Processing Error.

SOLUTION   None, call TIPS and send in the code. European customers should contact their local sales office.

LANGUAGES - OVL286 V3.2 **19**

This chapter lists known problems specific to the OVL286 product, Version 3.2. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

19.1   ERROR 301: Internal Processing Error

SYMPTOM   OVL286 issues ERROR 301: Internal Processing Error.

CAUSE   FORTRAN common blocks confuses OVL286.

SOLUTION   This problem will be fixed in the next release.

LANGUAGES - OH86 V1.0 20

This chapter lists known problems related to the OH86 product, Version 1.0. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

20.1   Output (HEX) File Sizes Differ Between iRMX® 286 and Series-III ISIS Systems

SYMPTOM   Output (HEX) file sizes differ between iRMX 286 and Series-III ISIS systems.

CAUSE   If there are conversion errors, the size of the HEX output file generated by iRMX 286-based OH86 may be, in some cases, shorter than the size of the file generated by the ISIS-based OH86.

SOLUTION   Avoid conversion errors.

LANGUAGES - PASCAL 286 V3.2 **21**

This chapter lists known problems related to the Pascal 286 language, Version 3.2. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

21.1 FATAL ERROR 9800 - Compiler Error In Code Generator

SYMPTOM A fatal error is issued from the code generator.

CAUSE Long and complicated statements exceed compiler limits. For example: a long statement (with more than one logical operator) can issue an error of this type.

```
b := (node[op2].fsym = nonterm) OR
     ((node[op1].fsym < node[op2].fsym) AND
     (node[op1].fsym <> nonterm));
```

SOLUTION Break the statement up by using temporary variables for intermediate results. For example:

```
xxx := (node[op2].fsym = nonterm);
yyy := (node[op1].fsym < node[op2].fsym);
zzz := (node[op1].fsym <> nonterm);
b := xxx OR ( yyy AND zzz );
```

21.2 Unexpected Results When Using Variant Records

SYMPTOM The fields of a variant record may have unexpected values if you are using OPTIMIZE(1) (the default) and depending on the different variants occupying the same memory space.

CAUSE When using OPTIMIZE(1), the code generator will not generate code to fetch a field of a record if that same field was previously accessed and is still residing in a register. Therefore, if you are hoping to alter bytes within words, etc. by using variant records and switching between the different variants, you may not get the results you expect. For example, if you assign a word to a variant record, use a different variant of the record to alter the upper byte of the word, then fetch the word from the record, you may get the original word with upper byte unchanged.

SOLUTION If OPTIMIZE(0) is used, you will get the results you want. If you must have OPTIMIZE(1), insert a dummy label or procedure call before any memory fetch that is getting "optimized out".

NOTE

Using variant records in this way is not legal in standard Pascal but the compiler cannot catch such usage without generating code to check hidden tag fields before accessing a variant record.

21.3 Problem Catching Stack Overflow in SMALL Model

SYMPTOM Your program may produce unexpected results when you are using the SMALL model and the stack overflows.

CAUSE The CHECK option does not detect stack overflow in the SMALL model of segmentation. The design of the SMALL model prevents reliable stack overflow detection.

SOLUTION Use the LARGE or COMPACT model of segmentation or do not depend on CHECK to catch stack overflow.

21.4 Function Declaration Problem

SYMPTOM A function that is declared with no parameters and no return type is treated as a procedure. No error is reported. Attempts to assign a return value to the function name will cause an error, as will attempts to use the function name in an expression.

CAUSE User program lacks a return type in function declaration.

SOLUTION This is a user error. Make sure that all function declarations include a return type.

21.5 Improper FILE Variable Initialization

SYMPTOM FILE variable does not get initialized properly.

CAUSE A FILE variable at the outer level of a program that is unknown to the main module will not get initialized properly. The compiler design does not permit such initialization.

SOLUTION This is a limitation caused by multi-module programs. Any global FILE variable must be made known to the main module via the FOR clause in the interface specification so that it can be initialized properly.

21.6 Floating-Point Problem

SYMPTOM An invalid RUN TIME INTEGER OVERFLOW ERROR is issued during run time.

CAUSE Some types of expressions that should be evaluated using floating-point arithmetic are not. For example:

```
REAL:= INTEGER + MAXLONGINT + 0.5
```

The first two operands on the right-hand side are computed using LONGINT arithmetic, causing an overflow.

SOLUTION This is not a problem. If you want to do floating-point calculations, make all operands REAL or use parentheses to cause the conversion as follows:

```
REAL:= INTEGER + (MAXLONGINT + 0.5)
```

21.7 TYPE Problem

SYMPTOM An error will not be generated if duplicate, identical TYPE declarations are made.

CAUSE User made a duplicate, identical type declaration in the same module and the compiler does not check for duplicate type definitions. For example:

```
T1 : INTEGER;
T2 : T1;
T2 : T1;
```

SOLUTION This is a user error. Do not make such declarations.

21.8 Case-Constant Problem Caused By Variant Record

SYMPTOM No extension message is issued or the non-existent case-constant may be used in an expression.

CAUSE Pascal does not detect an incomplete specification of tag case-constants in a variant record.

SOLUTION The compiler does not issue an error message when it should. The user can avoid this problem by checking to make sure case-constants in the variant record were specified completely.

21.9 CONST and TYPE Problem

SYMPTOM Multiple errors or no error may be generated when circular CONST or TYPE definitions are used.

CAUSE Circular CONST or TYPE definitions may cause multiple error messages (or no error message at all if defined in a module of the same name as the circular CONST or TYPE).

SOLUTION Do not make your CONST or TYPE definitions circular. For example:

```
------------------------------------
CONST x = y
      y = z
      z = x   ** circular definition **
------------------------------------
CONST x = y
      y = z
      z = 3   ** This is ok! **
------------------------------------
```

This is a user error and compiler problem. The compiler should display one correct error message.

21.10 NEW/DISPOSE Problem

SYMPTOM The compiler issues an error message when a literal negative constant is used in a NEW or DISPOSE statement.

CAUSE As a case-constant argument to NEW and DISPOSE, a named constant that is negative is allowed; however, a literal negative constant is flagged as an error (any sign, plus or minus, causes the parameter to become a non-constant).

SOLUTION Name (assign a name) your literal negative constant and use that named constant instead of using a literal negative constant directly.

21.11 NEW/DISPOSE Problem

SYMPTOM A wrong error message may be issued when using NEW/DISPOSE on an array that exceeds 8,388,608 bytes.

CAUSE Doing a NEW or DISPOSE on an array of 8,388,608 or more bytes gives a misleading error message "Expression too complex".

SOLUTION Dynamic arrays of this size are not supported by the compiler. If you require arrays of that size (only possible on 80286), you must use multiple smaller arrays.

21.12 TRUNC and ROUND Problem

SYMPTOM No extension message is generated by the compiler for incorrect usage.

CAUSE No extension message is generated when INTEGER arguments are passed to TRUNC and ROUND.

SOLUTION This is a user error, even though the compiler has not issued an extension message. Go back and check to make sure the program does not try to TRUNC or ROUND an INTEGER.

21.13 OUTBYT/OUTWRD Problem

SYMPTOM An unexpected result may be generated by the compiler.

CAUSE You cannot use OUTBYT/OUTWRD (port_addr, X) where X is an expression of a set type whose base type is an enumerated or subrange type of more than 16 objects. For example:

```
var S : set of Q
----------------------------------------
Q = (a1, a2, a3, .... a1b);
                     ** enumerated base type
P = a1 .. a3;
                     ** subrange
** This is fine
----------------------------------------
Q = (a1, a2, a3, ....a1b, a17)
                     ** more than 1b
P = a1 .. a3;

** When Q is > 1b, the compiler will let
   the Run-Time System do the set
   manipulations. The compiler does not
   correctly handle the set returned by
   the Run-Time System in this case.
----------------------------------------
```

SOLUTION Make X a variable, not an expression. If you need an expression, assign it to a temporary variable and then use the temp as the parameter. For example:

```
OUTWRD (PORT, S)          ** This is fine
            S       is a set variable.
OUTWRD (PORT, S1 + S2)    ** This is not.
            S1 + S2 is an expression.
```

21.14 Large-Array Problem

SYMPTOM An incorrect result may be generated by the compiler.

CAUSE Using WRITE(Large_Array_Char) where Large_Array_Char is a packed array of characters that is greater than 64K, does not produce correct results. There should be an error message when this is attempted.

SOLUTION This exceeds the compiler's limits. Use a loop to write the characters one at a time.

21.15   LARGE ARRAY Problem

SYMPTOM   LARGE ARRAYs may cause an unexpected result to be generated by the compiler.

CAUSE   A record cannot contain both LARGE ARRAYs and FILES. Any time a file or an array is in a record, the entire record will be initialized with zeros at initialization time. This destroys any selector tables used by LARGE ARRAYs in that same record.

For example:

```
R : record
    F : file of integer;
    A : array [ 1..100000 ] of char;
end;
```

```
        ** F and A are in a record, which will get
           initialized to 0.
        ** If A was not a  LARGE ARRAY i.e. less then
           64k-bytes, this would be fine.
        ** Because A is a  LARGE ARRAY, a selector
           table will be placed in the record
           following the field F.
        ** The initialization will set all of the
           record to 0, which will destroy the
           selector table.
```

SOLUTION   Do not put a LARGE ARRAY in a record with a file.

21.16   Pascal Run Time System Problem

SYMPTOM   A Pascal run time system error message is not generated when it is expected.

CAUSE   The Pascal run time system does not indicate an error when a text file is opened for random I/O.

SOLUTION   This is a user error. Do not attempt to open a text file for random I/O.

21.17   Pascal Run Time System Problem

SYMPTOM   The Pascal run time system gives a wrong error message.

CAUSE   The Pascal run time system will never generate the exception code 9105H (INTEGER Overflow On Input) when it should. Instead, it displays the code 9106H (LONGINT Overflow On Output).

SOLUTION   None at this time.

21.18   Run Time LIB (RTNULL.LIB) Problem

SYMPTOM   TQGetPreconnection may hang.

CAUSE   The preconnection list provided by TQDefaultPL in RTNULL.LIB is not set up correctly. This is used only when associating OUTPUT with :CO: and INPUT with :CI:.

SOLUTION   None at this time.

21.19   Array Indexing Problem In SMALL On 80286

SYMPTOM   Indexing into an array that has a non-zero lower bound may not work correctly on the 80286 in the SMALL model.

CAUSE   The binder produces incorrect fixups if the number of bytes between the zeroth index and the lower bound is greater than the number of bytes between the beginning of the segment and the beginning of the array when it is loaded into memory. This is a binder (OMF) problem, not a compiler error. This problem only occurs in the SMALL model.

SOLUTION   Do not use a non-zero lower bound.

21.20   ***FATAL ERROR 9803 IN (line number): COMPILER ERROR IN CODE GENERATOR

SYMPTOM   The compiler issues the following error message:

```
***FATAL ERROR 9803 IN (line number): COMPILER
ERROR IN CODE GENERATOR***
```

CAUSE   The compiler does not work with the functions PRED and SUCC of an unsigned 15-bit number.

SOLUTION   Use either of the following solutions:

1. Use a temp variable; assign the value to a temp variable and then use the temp variable as the argument for the built in functions PRED and SUCC.

   Example:

   ```
   program bug;
   type
      index = 1..525;
   var
      x,y : index;
      i,j : integer;
   begin
      i := x;
      y := pred(i);
      i := y;
      x := succ(i);
   end.
   ```

2. Do not use a subrange type that has the following attributes:

   Lower Bound   1<= lb <= 32767

   Upper Bound   256 <= ub <= 32767

   Examples:

LOWER BOUND	UPPER BOUND	RESULT
0	255	OK
0	525	OK
1	255	OK
1	256	ERROR
1	32767	ERROR
256	32767	ERROR

256	65535	OK

LANGUAGES - PL/M-86 V3.1 **22**

This chapter lists known problems related to the PL/M-86 language, Version 3.1. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

22.1 Reference To A Wrong Memory Location

SYMPTOM   Reference to wrong location in memory when referencing an absolute variable.

CAUSE   The bad reference happens only when referencing external variables that are declared: PUBLIC in another module, at an absolute location in memory, and initialized with the DATA attribute.

Example:

```
File 1.
        $LARGE
        m1: do;
        DECLARE x  WORD EXTERNAL;
        x = 5;        /* Reference wrong location in memory */
        END;

File 2.
        $LARGE
        m2: do;
        DECLARE x WORD PUBLIC AT(200h) DATA(32);
        END;
```

SOLUTION   Change any of these conditions. It is recommended to replace the DATA attribute in file 2 with the INITIAL attribute.

22.2 No Object File Produced

SYMPTOM   No object file is produced even though no errors or warnings were reported.

NOTE

On VAX, UDI-F-ABORT may be issued with no other warnings or errors.

CAUSE   The compiler failed to report an error. The error that would normally be issued is:

```
ERROR 199 LIMIT EXCEEDED: PROCEDURE COMPLEXITY FOR OPTIMIZE(2)
```

SOLUTION   Reduce the complexity of the procedure or use an optimization level less than two (i.e., OPTIMIZE(1)). This will be fixed in the next release.

22.3 Compiler Generates Bad Code

SYMPTOM   Compiler generates bad code when comparing pointers.

CAUSE   The bad code is generated only if the following conditions are true:

1. Each pointer is either an element of a based array of pointers or a member of a structure which is itself an element of a based array of structures.

2. Optimization level is 3.

SOLUTION   Use a lower optimizaton level or replace one of the subscripted pointers with a temporary non-subscript pointer.

22.4 Compiler Issues Error 118 "TYPE CONFLICT"

SYMPTOM   Compiler issues error 118 "TYPE CONFLICT"

CAUSE   This occurs when an index for an array is the result of an operation with two constants.

Example:

```
main: do;
DECLARE dx(3) BYTE, i BYTE;
    dx(i) = dx(2 XOR 3);
END;
```

SOLUTION   Use a temporary variable as the array index. This will be fixed in the next release.

Example:

```
main: do;
DECLARE dx(3) BYTE, i BYTE, temp BYTE;
    temp = 2 XOR 3;
    dx(i) = dx(temp);
END;
```

22.5 Type Mismatch Issued

SYMPTOM   Type mismatch is issued by Link-86.

CAUSE   The type mismatch happens with the MEDIUM ROM model of segmentation only when variables are initialized with the DATA attribute.

Example:

```
File 1.
        $medium rom
        m1: do;
        DECLARE a INTEGER, b INTEGER EXTERNAL DATA;
        a=b;
        END;

File 2.
        $medium rom
        m2: do;
        DECLARE b INTEGER PUBLIC DATA(15);
        END;
```

SOLUTION   Ignore the type mismatch.

22.6 ERROR 3 MISSING CONTROL PARAMETER Error

SYMPTOM "ERROR 3 MISSING CONTROL PARAMETER" error when subsystem definition is used.

CAUSE This occurs when the submodel of a subsystem definition is continued on next line the error is issued.

Example:

```
$compact(xyz
$-const in code- HAS main; EXPORTS test)
```

SOLUTION Place the submodel control on the same line as the subsystem name. This will be fixed in the next release.

Example:

```
$compact(xyz -const in code-
$HAS main; exports test)
```

22.7 Literally Defined Symbols Not Recognized

SYMPTOM Literally defined symbols are not recognized by the compiler.

CAUSE Under investigation.

SOLUTION Reduce the size of the literal.

22.8 Integer Value Between 32,630 And 32,767 Evaluated Incorrectly

SYMPTOM An integer variable containing a value between 32,630 and 32,767 is not evaluated correctly.

CAUSE The compiler will evaluate an integer between these two ranges as being negative.

SOLUTION This bug will be fixed in the next release of the compiler.

22.9 Bad Code Generated For Constants In A Reentrant Procedure

SYMPTOM Bad code is generated for constants that are declared local to a reentrant procedure.

CAUSE Constants that are declared local to a reentrant procedure are addressed incorrectly.

SOLUTION Constants should never be declared local to a reentrant procedure. An error will be reported in the next version of the compiler if this condition exists.

LANGUAGES - PL/M-286 V2.5 23

This chapter lists known problems related to the PL/M-286 language, Version 2.5. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

23.1 Local Variable Located Incorrectly

SYMPTOM Local variable in a factored list is located incorrectly if it is mixed in a factored declaration list along with the parameters.

Example:

```
TEST1: DO;
DECLARE STATIC (3) BYTE INITIAL (0,1,2);

             PROCA: PROCEDURE (A,B,C);
                    DECLARE X BYTE AT (@STATIC(0));
                    DECLARE (A,M,B,N,C) BYTE;
                    M = A + B;
                    N = B + C;
             END;

             CALL PROCA(10,20,30);

END;
```

CAUSE If the factored list immediately follows a local variable declared with the AT attribute and the list starts off with a parameter, the following locals in the list will not be located properly.

SOLUTION Do not declare a local variable with a parameter in the factored list.

23.2 Incorrect Code For FINDB

SYMPTOM The compiler generates incorrect code when using a FINDB built in function (with the same parameters and within the same block) the second time in the program.

Example:

```
TEST2: DO;

DECLARE BUF$OUT (100) BYTE INITIAL ('       %%%%%%%%');
DECLARE (X, Y, Z) WORD;

BUF$OUT (5) = '%';
Y = FINDB (@BUF$OUT,'%',SIZE(BUF$OUT));

BUF$OUT (Y) = ' ';
Z = FINDB(@BUF$OUT, '%',SIZE(BUF$OUT));

END;
```

CAUSE The compiler optimizes out completely the second FINDB built in function and assigns the value of Y to Z. Since both FINDB calls have the exact same parameters, the compiler thinks they must return the same value.

SOLUTION A workaround is to put a label on the second call or use optimize(0).

Example (Referring to the above mentioned program):

```
BREAK: Z = FINDB(@BUF$OUT, '%',SIZE(BUF$OUT));
```

23.3   Incorrect Code For A DWORD Expression

SYMPTOM   The compiler generates incorrect code for DWORD expression.

Example:

```
TEST3: DO;
DECLARE (X, Y) DWORD;
    P: PROCEDURE (B, D) DWORD;
            DECLARE B BYTE, D DWORD;
            RETURN B + D;
    END P;
Y = 5;
X = P (1, Y * 3 + 1);
END;
```

CAUSE   The compiler generates bad code in trying to re-use the constant 1, which was pushed onto the stack as a byte, but used again in the DWORD expression.

SOLUTION   A workaround to this problem is to put a second parameter of the procedure at a temporary location and use this temporary variable as a parameter.

Example:

```
TEST3: DO;

DECLARE (X, Y,TEMP) DWORD;

    P: PROCEDURE (B, D) DWORD;
            DECLARE B BYTE, D DWORD;
            RETURN B + D;
    END P;

Y = 5;
TEMP = Y * 3 + 1;
X = P (1, TEMP);

END;
```

23.4   Incorrect Code For Variable DECLAREd AT EXTERNAL

SYMPTOM   The compiler generates incorrect code when referencing a variable which is DECLAREd AT an EXTERNAL in another data segment.

Example:

```
$LARGE
TEST4: DO;

DECLARE X (100) BYTE EXTERNAL;
DECLARE ATX BYTE AT (@X(20));
DECLARE P POINTER;
DECLARE PB BASED P BYTE;

X (20) = 0;
ATX = 1;

P = @ATX;
PB = 1;

END;
```

CAUSE   Internal compiler error. The compiler assumes the variable is locally addressable, and does not generate indirect pointer references.

SOLUTION   Make changes in the above mentioned example as follows:

Example:

```
$LARGE
TEST4: DO;

DECLARE X (100) BYTE EXTERNAL;
DECLARE PTR POINTER;
DECLARE ATX BASED PTR BYTE;
DECLARE P POINTER;
DECLARE PB BASED P BYTE;

X (20) = 0;

PTR = aX(20);
ATX = 1;

P = aATX;
PB = 1;


END;
```

23.5 Variable DECLAREd AT Another Variable (ROM) Problem

SYMPTOM   If a variable A is declared AT variable B which is a constant in the CODE segment (compiled with ROM), then this ROM attribute is not propagated to A, the ATted Variable.

Example:

```
$ROM
TEST5: DO;
DECLARE DUMMY WORD;
DECLARE B BYTE DATA (2);
DECLARE A BYTE AT (aB);
A = 5;
END;
```

CAUSE   Internal compiler error.

SOLUTION   Avoid this kind of operation.

23.6 No Error Message

SYMPTOM   For Invalid Initialization The compiler does not report an error when the address of a dynamically allocated variable (such as a parameter) is used to initialize another variable.

Example:

```
TEST6: DO;
        PROC: PROCEDURE (PARM);
                DECLARE PARM BYTE;
                DECLARE PARM_PTR POINTER DATA(aPARM);

                PARM = 0;
        END;
END;
```

CAUSE   Internal compiler error.

SOLUTION   Do not use the address of a dynamically allocated variable (such as a parameter) to initialize another variable.

23.7 No Error Message For Too Many Values Initialization

SYMPTOM The compiler does not generate any error message when too many values are used to initialize a structure or array of structures.

Example:

```
TEST7: DO;

DECLARE B STRUCTURE (B1 BYTE, B2 BYTE) INITIAL (1, 2, 3);
DECLARE C (2) STRUCTURE (B1 BYTE, B2 BYTE) INITIAL
    (1,2,3,4,5);

END;
```

CAUSE The compiler does not issue ERROR 148 "INITIALIZING MORE SPACE THAN DECLARED," but it does issue ERROR 200 "ILLEGAL INITIALIZING OF MORE SPACE THAN DECLARED" later, if object is requested.

SOLUTION Do not initialize more space than declared.

23.8 No Error Message When Restricted Expression Is Used

SYMPTOM The compiler does not generate any error message when a restricted expression containing a real expression is used to initialize a real variable. Also, the expression is evaluated incorrectly.

Example:

```
TEST8: DO;

DECLARE R1 REAL INITIAL (1.1 + 2.2);
DECLARE R2 REAL INITIAL (2.2 - 1.1);
DECLARE R3 REAL INITIAL (0.0);

R3 = R2 + R1;

END;
```

CAUSE Internal compiler error.

SOLUTION Do not use real expressions in a restricted expression.

23.9 Initialized Character String Is Truncated

SYMPTOM The compiler does not generate any error message when using strings to initialize INTEGER, REAL, POINTER, and SELECTOR structure fields that follow BYTE, WORD, OR DWORD fields. Also, the string is silently truncated.

Example:

```
TEST9: DO;

DECLARE W STRUCTURE (B BYTE, I INTEGER) INITIAL ('ABC');
DECLARE X STRUCTURE (B WORD, R REAL) INITIAL ('ABCDEF');
DECLARE Y STRUCTURE (B DWORD, P POINTER) INITIAL ('ABCDEF');
DECLARE Z STRUCTURE (B BYTE, S SELECTOR) INITIAL ('ABC');

END;
```

CAUSE Internal compiler error.

SOLUTION Avoid these kinds of declarations.

23.10   Error 3 MISSING CONTROL PARAMETER Generated

SYMPTOM   The compiler generates Error 3 "MISSING CONTROL PARAMETER", when a subsystem control line consists only of a model and submodel.

Example:

```
$COMPACT(-CONSTANT IN CODE-)
```

CAUSE   Internal compiler error.

SOLUTION   Use the following controls to get the same effect.

```
$COMPACT
$ROM
```

23.11   No Error Message For Incorrect Restricted Expression

SYMPTOM   A restricted expression (INITIAL or AT expression) containing a variable reference as an array index is not detected as an error. The expression is evaluated incorrectly.

Example:

```
TEST11: DO;
        DECLARE A(21) WORD;
        DECLARE B WORD DATA(10);
        DECLARE C POINTER INITIAL(@A(B));
        .
        .
END;
```

CAUSE   Internal compiler error.

SOLUTION   Avoid this kind of operation.

23.12   Compiler Calculates Stack Size Incorrectly

SYMPTOM   If an exported procedure is called from a far procedure, the maximum stack size is calculated as if a near call is made instead of a far call, giving a stack value two less than it should be.

Example:

```
$COMPACT(OTHER EXPORTS CREATE)
$COMPACT(THIS ONE HAS TEST12; EXPORTS ROOT_P)

TEST12: DO;
CREATE: PROCEDURE(PARAM) EXTERNAL;
        DECLARE PARAM WORD;
END;
ROOT_P: PROCEDURE PUBLIC;
CALL CREATE(1);
END;
END;
```

CAUSE   Internal compiler error.

SOLUTION   A workaround to this problem is to increase the stack size during the link time using the SEGSIZE(STACK(..)) control.

23.13   Error 175 COMPILER ERROR: INVALID OPERATION Generated

SYMPTOM   The compiler generates "ERROR 175 COMPILER ERROR: INVALID OPERATION" when compiling the following code in any model except LARGE.

Example:

```
TEST13: DO;

DECLARE (I,X,F) BYTE;
DECLARE T(256) STRUCTURE (S(5) BYTE,A(5) BYTE) EXTERNAL;

T(I).S(F) = T(I).S(F) - T(X).S(F);

END;
```

CAUSE   Internal compiler error.

SOLUTION   A workaround to this problem is to use a temporary variable as shown below:

Example:

```
TEST13: DO;

DECLARE (I,X,F) BYTE;
DECLARE T(256) STRUCTURE (S(5) BYTE,A(5) BYTE) EXTERNAL;
DECLARE TEMP BYTE;

TEMP = T(I).S(F);
T(I).S(F) = TEMP - T(X).S(F);

END;
```

23.14   Compiler Generates Incorrect Code For Return Of Constant From DWORD Typed Procedure

SYMPTOM   The compiler generates incorrect code for the return of a constant from a DWORD typed procedure.

Example:

```
foo: do;
test: PROCEDURE (abc, def) DWORD PUBLIC REENTRANT;
          DECLARE abc selector,
                  def pointer;
          RETURN (0);
end test;
end foo;
```

CAUSE   Internal compiler error.

SOLUTION   Assign a local DWORD variable to the value of the constant and return the variable instead.

Example:

```
foo:  do;
test:PROCEDURE (abc,def) DWORD PUBLIC REENTRANT;
       DECLARE abc selector;
       DECLARE def pointer;

       DECLARE lmn dword;
              lmn = 0;
       RETURN (lmn);
```

```
end test;
end foo;
```

23.15 Zero Length DATA Segment Is Created

SYMPTOM   A zero length DATA segment is created.

CAUSE   If the $LARGE and $RAM controls are in effect, a data segment of exactly 65535 bytes is considered by the compiler to be of zero length. The following example will create a zero length data segment:

Example:

```
$LARGE RAM
BUG: DO;
     DECLARE ARRAY(65535) BYTE;
END;
```

SOLUTION   Insure that less than 65535 (0FFFFH) bytes are consumed in the DATA segment.

23.16 Constants Are Not Referenced Properly When Using $LARGE And $ROM

SYMPTOM   Constants are not referenced properly when using $LARGE and $ROM compiler controls simultaneously.

CAUSE   Internal compiler error.

When using the $LARGE and $ROM compiler controls simultaneously, the compiler generates incorrect references to constants in the CODE segment which are more than 07FH away from the beginning of the constants.

The occurrence of the problem is calculable, but is governed by the number of constants and the number of implicit pointers which the compiler needs to generate to reference them.

This may appear to be an incorrectly generated MODRM byte.

SOLUTION   Avoid this type of situation.

23.17 Compiler Reports ERROR 35 When Using $INCLUDE

SYMPTOM   The compiler reports ERROR 35 when using $INCLUDE.

CAUSE   Internal compiler error.

When compiling modules that contain nested $INCLUDE controls, the compiler randomly generates ERROR 35 (LINE GREATER THAN 128 CHARACTERS). This problem has been reproduced when the $INCLUDE controls are nested.

SOLUTION   Try recompiling or restructuring your application to overcome this problem.

23.18 Compiler Reports ERROR 205

SYMPTOM   Under some circumstances, the compiler will report ERROR 205, "BAD CODE GENERATED."

CAUSE   Internal compiler error. The compiler attempted to optimize code and did so incorrectly.

SOLUTION   Place a label at the statement that generated the error. Alternatively, use optimize level 0.

23.19   Debug Information Is Incompatible With I2ICE

SYMPTOM   Debug information generated by the compiler is incompatible with I2ICE.

CAUSE   The compiler does not place into the object module the correct debug symbol information that I2ICE expects under the following conditions:

1. A variable is based on a structure element, e.g.,

```
DECLARE A_S STRUCTURE (A_PTR POINTER);
DECLARE A1 BASED A_S.APTR BYTE;
```

2. Within a block, the declaration of a variable occurs AFTER a nested block.

SOLUTION   PL/M-286 V2.6 (available on ISIS/iNDX and PC-DOS) corrects this problem.

23.20   Application Compiles But You Get TYPE MISMATCH Warnings When BiNDing

SYMPTOM   Your application compiles fine, but you get TYPE MISMATCH warnings when BiNDing.

CAUSE   Internal compiler error. When using $LARGE and $ROM compiler controls simultaneously, the compiler generates incompatible TYPEDEFS for constants which have been declared as PUBLIC in one module and EXTERNAL in another.

SOLUTION   Avoid using both the $LARGE and $ROM controls together or alter your use of constants.

23.21   MOVB, MOVW, MOVRB and MOVRW Not Working

SYMPTOM   MOVB, MOVW, MOVRB, and MOVRW may not work correctly in some cases when using OPTIMIZE(1), OPTIMIZE(2), or OPTIMIZE(3).

Example:

```
$LARGE

bug: DO;

DECLARE (a$ptr, b$ptr) POINTER EXTERNAL;
DECLARE a$descr(10) SELECTOR EXTERNAL;
DECLARE b$descr(10) SELECTOR EXTERNAL;
DECLARE (index, frame, desc$no) BYTE EXTERNAL;

DO index = 0 TO 3;
    desc$no = (frame - 1) * 4 + index;
    a$ptr = BUILD$PTR(a$descr(2),0);
    b$ptr = BUILD$PTR(b$descr(desc$no),0);

/*** fix_the_bug:   ***/
    CALL MOVW(a$ptr, b$ptr, 10H);

END;

END bug;
```

CAUSE   Due to incorrect optimization, the DS register is not set up correctly for the procedure.

SOLUTION   Remove the comments around the label in the example above.

23.22 IF-THEN Comparisons Sometimes Ignored

SYMPTOM In the following example, the compiler optimizes out the comparison between b and 0, so a is not set correctly.

Example:

```
test: DO;

DECLARE a BYTE, b BYTE, c BYTE;

a = 5;
b = 2;

/*** add_label:  ***/
IF b > 0
    THEN a = 1;

c = 2;
END;
```

The following conditions must be met to have the compare optimized out:

1. The comparison must be against 0 (zero).

2. The test must NOT be for equivalence.

3. The variable being tested must be set close to the test. This allows the compiler to "know" and substitute the value of the variable for the variable.

CAUSE Internal compiler error.

SOLUTION Remove the comments around the label in the example above.

23.23 ERROR 118: TYPE CONFLICT Issued When Constants Used In Array Expression

SYMPTOM The compiler issues a type conflict message when a logical expression with constants is used as an array subscript.

Example:

```
test: DO;

DECLARE digit(10) BYTE;
DECLARE name(10) BYTE;

name(6) = digit(35H AND 0FH);

END test;
```

CAUSE This is a coding error. There is a type mismatch. According to the rules of PL/M, an array subscript may be an INTEGER, BYTE, or WORD variable. When a constant expression is used as a subscript of an array variable, it is evaluated as if it were an INTEGER expression. This means that the constants 35H and 0FH are treated in a signed context as INTEGERS. Boolean operators (i.e., NOT, AND, OR, and XOR) can only be used with BYTE, WORD, or DWORD values. The type conflict occurs because the constants, which are treated as INTEGERS, cannot be part of a boolean expression.

SOLUTION   Force the compiler to treat the constants as byte values.

Examples:

```
/*** Use a temporary BYTE, WORD, or DWORD
       variable to hold the boolean expression.  ***/

DECLARE i BYTE;

i = 35H AND 0FH;
name(b) = digit(i);
       .
       .
       .
/*** Or, use one of the compiler built-in
       functions to explicitly state the data
       type of the array index.                   ***/

name(b) = digit(LOW(35H AND 0FH));
```

23.24 General Protection Error

SYMPTOM   A general protection error is received when referencing byte variables at the end of a dynamically allocated segment.

CAUSE   When the last byte in a segment is coerced into a word (for example, when it is pushed onto the stack or used in an OUTWORD function), the second byte is past the end of the segment. When the processor checks the segment limit, it gives a general protection error.

           This only affects dynamically allocated segments since the binder and builder pad statically allocated segments to avoid this problem.

SOLUTION   Allocate an extra byte at the end of a dynamically allocated segment when the last byte will be pushed onto the stack.

23.25   Bad Addresses Generated by Using Based Variables As Array Subscripts

SYMPTOM   The compiler may incorrectly optimize statements that use far pointers. If a far pointer is assigned the address of an array element and the array subscript is a based variable, in a subsequent reference to the address of another element in the same array, the address is formed incorrectly. In the example below the address of the first parameter to the call is incorrect.

Example:

```
$LARGE
test:DO;

proc_a:PROCEDURE (out_ptr,size_ptr) PUBLIC;
    DECLARE out_ptr POINTER, size_ptr POINTER;
END proc_a;

DECLARE a_ptr POINTER;

proc_b:PROCEDURE (output_ptr, word_ptr) PUBLIC;

DECLARE output_ptr POINTER;
DECLARE word_ptr POINTER;
DECLARE out_array BASED output_ptr(1) BYTE;
DECLARE a_word BASED word_ptr WORD;

a_ptr = @out_array(a_word);
/*** fixit: ***/
CALL proc_a(@out_array(0),@a_word);

END proc_b;
END test;
```

CAUSE   Incorrect optimization by the compiler at optimization levels 1, 2, and 3.

SOLUTION   Place a label after the first reference to the address of the array. In the example above, remove the comments around the label. Alternatively, use optimize level 0. This problem will be fixed in a future release of the compiler.

23.26   Incorrect Optimization When STACKPTR Used in Arithmetic Expressions

SYMPTOM   When the PL/M built-in variable STACKPTR is used in arithmetic expressions, and these arithmetic expressions are the same, the calculations in the second and subsequent expressions are not performed. This error occurs at all optimization levels except OT(0).

Example:

```
test:DO;
    STACKPTR = STACKPTR - 2;
    /*** other statements ***/

    /*** label1: ***/
    STACKPTR = STACKPTR - 2;
END test;
```

CAUSE   Internal compiler error.

SOLUTION   Use optimization level 0 or add a label at the second and subsequent statements. In the example above, remove the comments from the label.

23.27   Problem With Adding in PL/M Data Statements

SYMPTOM   Unexpected results when adding constants together in a restricted expression in the PL/M-n86 compilers.

Example:

```
DECLARE a$var WORD DATA(240+21);

A$var will be initialized to 5 ((240+21) MOD 256) and not 261.
```

CAUSE   When a constant expression is evaluated, the type of variable is dictated by the type of the first constant. Anytime the sum of the added constants overflows the maximum value for the first data type, the sum of the expression will be the expected sum MOD one plus the maximum value for the data type. In the example above the actual value (5) is the expected sum (261) MOD one plus the maximum value for the data type (1+255).

SOLUTION   When the data type of the sum of the values will exceed the limit of the data type of the first element, add and subtract a value which will cause the compiler to treat the value correctly.

Example:

```
DECLARE a$var WORD DATA (256-256 + 240 + 21);
```

23.28   ERROR 47: LIMIT EXCEEDED: PROCEDURES (253)

SYMPTOM   The PL/M-286 compiler issues ERROR 47, "LIMIT EXCEEDED: PROCEDURES (253)" when the number of procedures apparently does not approach the limit.

CAUSE   All procedure declarations count - even EXTERNAL procedures in INCLUDE files. The iRMX 286 Release 2 system INCLUDE file RMXPLM.EXT contains a large number of external declarations. If this file is used, the number of other user procedures which can be declared in a PL/M-286 module is about fifty. The table below shows the iRMX 286 Release 2 INCLUDE file names and the procedure count for each file.

```
BIOS.EXT        36
EIOS.EXT        31
HI.EXT          17
LOADER.EXT       6
NUCLUS.EXT      68
RMXPLM.EXT     192
UDI.EXT         34
```

SOLUTION   Since the RMXPLM.EXT file contains the external definitions of BIOS.EXT, EIOS.EXT, HI.EXT, LOADER.EXT, NUCLUS.EXT, and UDI.EXT, you can reduce the number of external declarations by using INCLUDE files for only the iRMX 286 layers needed. This limit will be raised in a future version of the PL/M-286 compiler.

TOOLS AND UTILITIES - AEDIT 286 V2.1   24

This chapter lists known problems specific to AEDIT-286, Version 2.1. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

24.1 Lines Greater Than 255 Characters Displayed Erroneously

SYMPTOM   The display of lines which are greater than 255 characters is sometimes erroneous.

CAUSE   A problem in the AEDIT286 editor.

SOLUTION   The only solution is to keep all lines less than 255 characters in length.

24.2 AEDIT286 "!SYSTEM" Command Causes System To Crash

SYMPTOM   The AEDIT286 "!SYSTEM" command causes the system to crash.

CAUSE   Using the AEDIT286 "!SYSTEM" command to activate the iRMX® 286 "attachfile" command may cause the system to crash.

SOLUTION   Do not attempt to use the AEDIT286 "!SYSTEM" command to change directories.

24.3 BLOCK Marker ('@',40H) Does Not Disappear After BLOCK Command Completes

SYMPTOM   The BLOCK marker ('@',40H) does not disappear after the BLOCK command completes.

CAUSE   In some cases, when the BLOCK command is used on lines greater than 80 characters, the BLOCK marker ('@',40H) does not disappear after the BLOCK command is completed.

SOLUTION   You must refresh the screen explicitly using the V command.

24.4 UNDO Function Does Not Work On iRMX 286 Systems

SYMPTOM   The UNDO function, set to ^U (15H) by default, does not work on iRMX 286 systems.

CAUSE   The ^U code is processed by the Human Interface layer and is not passed to the AEDIT286 editor.

SOLUTION   You must set UNDO to another input code in your AEDIT.MAC file (e.g., set it to ^Y by defining AFXU=19) or use HI commands to allow ^U to be passed to the AEDIT286 editor.

24.5 General Protection Exception When Saving A File Using AEDIT

SYMPTOM   A General Protection exception may be encountered when saving a file on a floppy using AEDIT.

CAUSE   The floppy was removed from the diskette drive. The problem is under investigation.

SOLUTION   Ensure that a floppy diskette is in the drive and the floppy drive door is closed.

TOOLS AND UTILITIES - NUMERICS V3.0

25

This chapter lists known problems related to the iRMX 286 version of NUMERICS, Version 3.0. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

25.1 System Hangs When Performing Calculations

SYMPTOM   The system hangs when performing calculations using complex numbers.

CAUSE   When performing calculations using complex numbers, the error handler must not be allowed to handle exceptions. The 80287 chip itself must be allowed to handle any complex number exceptions, whether the way it handles the exception is ideal or not. If a complex number exception occurs and the exception is not masked, the system will "hang" and your only recourse will be to reset it.

SOLUTION   Use the following PL/M commands to mask or unmask the exception (a masked exception is automatically handled by the 80287 chip):

```
call set$real$mode (03FFH);    /* mask   */

call set$real$mode (0300H);    /* unmask */
```

The PL/M set$real$mode procedure is described in section 10.7 of the *PL/M-286 User's Guide.*

TOOLS AND UTILITIES - 8087 NUMERICS SUPPORT 26

This chapter lists known problems related to the iRMX 286 version of 8087 Numerics Support. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

26.1 Long Real and Temp Real Parameters Result in Erroneous Data

SYMPTOM When calling functions of the DCON87 and DC287 libraries with long real and temp real parameters, erroneous data results.

CAUSE In Chapter 3 of both the 8087 and 80287 Numerics Support documention, the constant definitions for long real and temp real are incorrectly noted as 2 and 3, respectively.

SOLUTION The correct values for long real and temp real are 1 and 2, respectively.

TOOLS AND UTILITIES - 80287 NUMERICS SUPPORT **27**

This chapter lists known problems related to the iRMX 286 version of 80287 Numerics Support. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

27.1 Long Real and Temp Real Parameters Result in Erroneous Data

SYMPTOM When calling functions of the DCON87 and DC287 libraries with long real and temp real parameters, erroneous data results.

CAUSE In Chapter 3 of both the 8087 and 80287 Numerics Support documention, the constant definitions for long real and temp real are incorrectly noted as 2 and 3, respectively.

SOLUTION The correct values for long real and temp real are 1 and 2, respectively.

TOOLS AND UTILITIES - Soft-Scope* 286 DEBUGGER V2.55

28

This chapter lists known problems related to the iRMX 286 version of Soft-Scope 286, Version 2.55. Each problem contains a brief description of the symptom, its cause, and solution.

New entries, or previously published problems whose status has changed, will be identified by change bars.

28.1   Soft-Scope 286 Debugger Hanging Due To No 80287 Chip

SYMPTOM   The Soft-Scope 286 debugger either hangs or causes an "Interrupt 7" before signing on.

CAUSE   The Soft-Scope 286 debugger tries to initialize the non-existent 80287 chip with an FNINIT instruction and a fault is caused.

SOLUTION   Create a file called "SSCOPE.NPX" in the same directory as the Soft-Scope 286 debugger (it can be empty). This will cause the Soft-Scope 286 debugger to skip the initialization of the 80287 chip. In this case, no floating-point displays will be available. This problem is documented at the end of Chapter 2 in the *Soft-Scope 286 Source-Level Debugger User's Guide.*

28.2   Application With Overlays Will Not Load

SYMPTOM   "Overlays not supported" message given.

CAUSE   Current version does not support overlays.

SOLUTION   The problem will be fixed in a future release of the Soft-Scope 286 debugger.

28.3   "STACK" Command Does Not Work Properly In User-Created Tasks

SYMPTOM   The STACK command does not work properly in user-created tasks (not an initial task).

CAUSE   The iRMX 286 operating system does not clear or initialize a newly created task's stack.

SOLUTION   When creating a task, pass a stack segment in the create task system call. Before making the call create a segment via RQ$CREATE$SEGMENT and fill in the segment with "C7H".

28.4   Incorrect Source Lines Displayed While Using STEP In PL/M Modules

SYMPTOM   The Soft-Scope 286 debugger seems to be a few lines "off" in its display of current line.

CAUSE   When using PL/M at optimization levels greater than 0, the line numbers in the code listing may not directly correspond to the actual object code. This is especially evident near the ends of nested blocks (for example, "DO WHILE" blocks.)

SOLUTION   Put a label at the first executable line after the end of the nested blocks or use optimize level 0.

28.5   Soft-Scope 286 Debugger Seems "way off" In Report Of Current Line Number

SYMPTOM   STEP or LINE reports a line far away from actual current line, reporting it "near" a line when, in fact, it is at a completely different one.

CAUSE   External procedure declarations are declared after non-external declarations.

SOLUTION   Re-order procedure declarations so all external declarations come first.

28.6   General Protection Error After Soft-Scope 286 Debugger Exit

SYMPTOM   A general protection error is displayed after a Soft-Scope 286 debugger exit.

CAUSE   The stack exception (INT 12) is caused by a user-created task while the Soft-Scope 286 debugger is currently debugging another task.

SOLUTION   Run the application with no breakpoints to find the area with the stack problem.

CAUSE   DQEXIT or RQEXITIOJOB called by a user-created task while the Soft-Scope 286 debugger is still with another task in the foreground.

SOLUTION   Have the main task be responsible for all EXITs.

28.7   Multiple-Dimensioned Arrays Displayed Incorrectly In FORTRAN

SYMPTOM   Reference such as ".XYZ(3,2,1)" displays contents of XYZ(3,2,1) in FORTRAN.

CAUSE   FORTRAN stores arrays in row-minor order rather than row-major.

SOLUTION   This is normal FORTRAN operation.

28.8   General Protection Exception When Exiting Soft-Scope 286 Debugger

SYMPTOM   While one user is using the Soft-Scope 286 debugger at one terminal, a second user runs an application which causes a general protection fault (INT 13), or stack fault (INT 12), or double fault (INT 8). This causes the second user's system to hang, and the Soft-Scope 286 user will get a general protection fault after the Soft-Scope 286 debugger exits.

CAUSE   The Soft-Scope 286 debugger is set up as the handler for these traps, but cannot report the trap if it occurs in another user's job.

SOLUTION   Ensure that undebugged applications are executed under control of the Soft-Scope 286 debugger.

SOFTWARE PRODUCT CURRENT VERSION LISTS **A**

This section contains a list of the current versions for the software products supported by Intel's Technical Support Operation. These products are categorized by the environments covered by our Software Support Contracts (SSCs). For information about obtaining an SSC, refer to page 1 of the Preface.

Changes or additions to this current version list are noted by bar marks in the right-hand column.

Current Version List — Q1/1990

SSC 100
Applies to PDS/MDS 800/S-II 8-Bit Environment

Product	Description		Version
FTRANS	PDS-FTRANS		1.0
ICE 42	ICE 42		1.1
ICE 44	ICE 44		1.0
ICE 49A	ICE 49A		1.2
ICE 51	ICE 51		2.0
ICE 80	ICE 80		4.0
ICE 85B	ICE 85B		2.5
ICE 86A	ICE 86A		2.2
ICE 88A	ICE 88A		2.2
IPPS	IPPS (8-bit)		1.4
ISIS	ISIS II (MDX2100)		4.3
	ASM 80/85	4.1	
MCI51ASM	ASM51, RL51, LIB51, CONV51 (8-bit)		2.2
	ASM51	2.1	
	RL51	3.0	
	LIB51	1.0	
	CONV51	1.0	
	UTIL	1.1	
MDS 301	FORTRAN 80		2.1
MDS 308	ASM86		2.1
	OH86	1.2	
	LIB86	1.2	
	UPM	3.0	
	LINKER	1.3	
	LOCATOR	1.3	
	CONV86	2.0	
MDS 311	PL/M-86 (Includes ASM 86 V2.1)		2.1
	OH86	1.2	
	LIB86	1.2	
	UPM	3.0	
	XREF	1.2	
	LINKER	1.3	
	LOCATOR	1.3	
	CONV86	2.0	
	COMPILER	2.1	
	87 EMULATOR	1.1	
	87 NUM. LIB	1.1	
MDS 312	ASM89		2.0
	LINKER	1.3	
	LOCATOR	1.3	
	OH86	1.2	
	UPM	3.0	
	ASSEMBLER	2.0	
MDS 318	FUND. SUPPORT PACKAGE		1.0
MDS 320	BASIC 80		1.1
MDS 360	CREDIT		2.0
MDS 380	COBOL		1.0
MDS 381	PASCAL 80		2.1

Product	Description		Version
MDS 384	MAINFRAME LINK		1.1
MDS 387	ASM 86/87/88		3.0
	ASSEMBLER	2.1	
	LIB/EMULATOR	1.2	
MDS D48	ASM48		4.2
MDS PLM	PL/M-80		4.0
	COMPILER	4.0	
	LIB/EMULATOR	1.2	
MDX 335	AEDIT 80		2.0
MDX 352	PL/M-51, RL51, LIB51 (8-bit)		1.1
	PL/M-51	1.1	
	RL51	3.0	
	LIB51	1.0	
	UTIL51	1.1	
MDX 2001	ISIS III N (8-bit)		2.2
MDX 2009	ISIS IIW (MDX750)		4.3W
PDS 100	ISIS-PDS		1.1
PDS EMV51	EMV 51		1.0
PDS EMV51A	EMV 51A		1.1
PDS EMV88	EMV 88		1.0

SSC 200
Applies to S-III/S-IV 16-Bit Environment

Product	Description		Version
ICE 42	ICE 42		1.1
ICE 44	ICE 44		1.0
ICE 49A	ICE 49A		1.2
ICE 51	ICE 51		2.0
ICE 80	ICE 80 (Series III only)		4.0
ICE 85B	ICE 85B		2.5
ICE 86A	ICE 86A		2.2
ICE 88A	ICE 88A		2.2
III 902B/903C	086/088 PROBE SOFTWARE		1.9
III 912B/913C	186/188 PROBE SOFTWARE		1.9
III 923B/923C	286 PROBE SOFTWARE		1.9
III 953B/953C	HOST SOFTWARE		1.9
III 981	ILTA SOFTWARE		1.8
IPPS	IPPS - S III		1.4
IPPS	IPPS - S IV		2.2
ISIS	ISIS II (MDX2100)		4.3
	ASM80/85	4.1	
I86ASM51	ASM51 SINGLEUSER (16-bit)		2.4
	ASM51	2.3	
	LIB51	1.1	
	RL51	3.1	
	OH	1.1	
I86C86	iC-86 SINGLEUSER (MWC)		3.0
I86C286	iC-286 SINGLEUSER (MWC)		3.0
I86PLM51	PL/M-51 SINGLEUSER (16-bit)		1.3
	LIB51	1.1	
	PL/M-51	1.3	
	RL51	3.1	
	OH	1.1	
I86PLM86	PL/M-86 SINGLEUSER (MDS313)		3.1
I86PAS86	PASCAL 86 (MDS314)		3.1
I86FOR86	FORTRAN 86 (MDS315)		3.0
MDS 319	87 LIB		3.0
MDS 320	BASIC 80 (8-bit)		1.1
MDS D48	ASM48 (8-bit)		4.2
I86ASM86	iAPX86 S/W DEV PKG (MDX316)		4.1
	ASM86	2.1	
	LINK86	2.7	
	LOC86	2.5	
	LIB86	2.1	
	OH86	1.0	
	CREF86	1.0	
I86ASM286	ASM286 S/W DEV PKG (MDX321)		3.2
	ASM286	1.3	
	BND286	3.2	
	BLD286	3.2	
	LIB286	3.2	
	MAP286	3.2	

Product	Description		Version
	OVL286	3.2	
MDX 322	APX286 EVAL PKG 1.0		
	ASSEMBLER	1.0	
	BUILDER	1.0	
	SIMULATOR	1.0	
	RUN-TIME SUPPORT	1.0	
	DATA Def.	1.0	
I86PLM286	PL/M-286 (MDX323)		3.1
I86PAS286	PASCAL 286 (MDX324)		3.1
MDX 329	287LIB 3.0		
MDX 332	PMT		2.1
I86PSC86	PSCOPE (MDX333)		1.1
I86EDI86	AEDIT 86 (MDX334)		2.0
MDX 337	E-MAIL		1.0
I86ASM96	ASM96 (MDX355)		1.2
	ASM96	1.2	
	RL96	2.2	
	LIB96	1.2	
	FPAL96.LIB	2.2	
	OH	1.1	
I86PLM96	PL/M-96 (MDX356)		1.2
	PL/M-96	1.2	
	RL96	2.2	
	LIB96	1.2	
	FPAL96.LIB	2.2	
MDX 394	ACL/VMS		2.0
MDX 395	ACL/UNIX		2.0
MDX 2002	SERIES IV (MDX430)		3.2
MDX 2005	NDS II (MDX450) (16-bit)		3.2
SBE 96	SBE 96		2.0
SIII	SERIES III (MDX2101)		4.3
	ASM 86/88	2.0	
TSCOPE186	TSCOPE 186		1.1
VLSiCE-96	VLSiCE-96		3.0
iPAT86ISS	iPAT86ISS		1.2
iPAT86NDX	iPAT86NDX		1.2
MDX555	INDX OPENNET SOFTWARE		3.2
MDX465	OpenNET NRM SOFTWARE		3.2
MDX485	COMPILE ENGINE		3.2
SA252	ICE 5100/252		3.0
SA452	ICE 5100/452		3.0
SA044	ICE 5100/044		3.0

SSC 300
Applies to Intel's VAX/VMS Resident Cross Software

Product	Description		Version
VVSASM86	ASM86VX (MDX341VX)		4.1
	ASM86	3.2	
	CREF86	1.3	
	LINK86	2.9	
	LOC86	2.8	
	LIB86	2.4	
	OH86	1.3	
VVSPLM86	PL/M-86 (MDX343VX)		3.3
VVSPAS86	PASCAL 86-VX (MDX344VX)		3.3
VVSASM286	ASM286-VX (MDX371VX)		3.2
	ASM286	1.3	
	BND286	3.2	
	BLD286	3.2	
	MAP286	3.2	
	LIB286	3.2	
	OVL286	3.2	
VVSPLM286	PL/M-286-VX (MDX373VX)		3.3
VVSPLM386	PL/M-386-VX		3.3
VVSASM386	ASM386VX		4.0
MDX 392	NDS VAX LINK		3.0
MDX393F	NDS VAX LINK (RX-50)		2.1
MDX393T	NDS VAX LINK (TK-50)		2.1
MDX 394	ACL-VMS		2.0
MDX 395	ACL-UNIX		2.0
VVS C86	iC-86-VX (MDX347VX) (MWC)		3.0
VVS C286	iC-286-VX (MDX377VX) (MWC)		3.0
VVS C386	iC-386-VX		1.2
VVSRLL386	386 R & L TOOLSET		1.3
	BLD386	1.5	I
	BND386	1.4	I
	LIB386	1.2	I
	MAP386	1.2	I
VVSFORT86	FORTRAN 86		3.0
VVSFORT386	FORTRAN 386		4.0
C960V	iC960 Compiler		1.2
ASM960V	80960 Assembler & Utilities		1.2
	ASM960	1.2	
	ARC960	1.2	
	DIS960	1.2	
	DMP960	1.2	
	LNK960	1.2	
	MPP960	1.2	
	NAM960	1.2	
	ROM960	1.2	
	SIZ960	1.2	
	STR960	1.2	
MVVSASM86	ASM 86 MICRO VAX		3.2 I
MVVSASM386	ASM386VX MICRO VAX		4.0

Product	Description	Version	
MVVSPLM86	PL/M-86 MICRO VAX	3.3	
MVVSC86	iC-86 MICRO VAX	3.0	
MVVSFORT86	FORTRAN 86 MICRO VAX	3.0	
MVVSFORT386	FORTRAN 386	4.0	
MVVSASM286	ASM286 MICRO VAX	1.3	I
MVVSC286	C286 MICRO VAX	3.0	
MVVSC386	C386 MICRO VAX	1.2	
MVVSRLL386	RLL386 MICRO VAX	1.3	
MVVSPLM286	PL/M-286 MICRO VAX	3.3	
MVVSPLM386	PL/M-386 MICRO VAX	3.3	
VVSKIM	NVL Modification	1.0	
	— to install VAXLINK on VMS 4.7 & 5.0		

Note: Where a MVVS (MICRO VAX) product is listed, it differs only in media type (TK50 Cartridge Tape) from its VVS counterpart, which is distributed on reel tape. The software contained on both media is otherwise identical.

C960M	TK-50 Distribution Of C960V For Micro Vax	1.2
ASM960M	TK-50 Distribution Of ASM960V For Micro Vax	1.2
EVA960KB	80960 Execution Vehicle & SDM (PC-DOS-hosted)	1.2

SSC 400
Applies to the iRMX I Operating System

Product	Description		Version	
DCX 51	DISTRIBUTED CONTROL EXECUTIVE		2.0	
PCS86RMX	PSCOPE 86		2.0	
R86C86	iC-86 SINGLE-USER (RMX866)		4.1	
R86ASM86	ASM86 (RMX860)		4.1	
	ASM86	2.1		
	CREF86	1.0		
	LINK86	2.7		
	LOC86	2.5		
	LIB86	2.1		
	OH86	1.0		
R86PLM86	PL/M-86 SINGLE-USER (RMX863)		3.1	
RMX I	iRMX I OPERATING SYSTEM		8.0 UP1	I
RMX 86 HAN	iRMX I HANDYMAN KIT		1.0	
	R86EDI86	2.0		
R86PAS86	PASCAL 86 (RMX861)		3.4	I
R86FOR86	FORTRAN 86 (RMX862)		3.0	I
R86EDI86	AEDIT 86 (RMX864)		2.0	
SDM	iSDM SYSTEM DEBUG MONITOR		3.2	
SYR 86	iRMX SYSTEM LANGUAGE KIT		7.0	
	R86ASM86	4.1		
	R86EDI86	2.0		
	R86PLM86	3.1		
VDI 720	GRAPHICS INTERPRETER		1.8	
DCS100	BITBUS TOOLBOX		2.0	
DCS110	BITWARE		2.0	
DCS120	PROGRAMMER SUPPORT PACKAGE		2.0	
R86ASM51	ASM51 SINGLE-USER (16-bit)		2.4	
	ASM51	2.3		
	LIB51	1.1		
	RL51	3.1		
	OH	1.1		
R86PLM51	PL/M-51 SINGLEUSER (16-bit)		1.3	
	LIB51	1.1		
	PL/M-51	1.3		
	RL51	3.1		
	OH	1.1		
PLM96	PL/M-96		1.2	

SSC 420
Applies to the iRMX II Operating System

Product	Description		Version
RMX II	iRMX II OPERATING SYSTEM		4.0 UP1 I
R286ASM286	ASM 286 - ASSEMBLER		3.2
	ASM286	1.3	
	BND286	3.2	
	BLD286	3.2	
	LIB286	3.2	
	MAP286	3.2	
	OVL286	3.2	
R286EDI286	AEDIT286		2.2
R286FOR286	FORTRAN 286		3.1
R286PAS286	PASCAL 286		3.2 I
R286PLM86	PL/M-86		3.1
R286PLM286	PL/M-286		3.3
R286C286	iC-286		4.1
R286ASM86	ASM86 - ASSEMBLER		4.1
	ASM86	3.2	
	LINK86	2.7	
	LOC86	2.5	
	LIB86	2.1	
	CREF86	1.0	
	OH86	1.0	
SDM	iSDM SYSTEM DEBUG MONITOR		3.2
SYRII	iRMX II SYSTEM LANGUAGE KIT		2.0
	R286ASM86	4.1	
	R286ASM286	3.2	
	R286PLM286	3.1	
	R286EDI286	2.1	
RMXIISFTSCP	Soft-Scope II DEBUGGER(R286SSC286)		2.8
DCS100	BITBUS TOOLBOX		2.0
DCS110	BITWARE		2.0
DCS120	PROGRAMMER SUPPORT PACKAGE		2.0
VDI720	GRAPHICS INTERPRETER		1.8
RMXSCS	iRMX SOURCE CONTROL SYSTEM		1.0
RMX286TLB	iRMX TOOLBOX		2.1
R86ASM51	ASM51 Single-User (16-bit)		2.4
	ASM51	2.3	
	LIB51	1.1	
	RL51	3.1	
	OH	1.1	
R86PLM51	PL/M-51 Single-User (16-bit)		1.3
	LIB51	1.1	
	PL/M-51	1.3	
	RL51	3.1	
	OH	1.1	

SSC 430
Applies to the iRMK Real-Time Kernel

Product	Description	Version
RMK	iRMK Kernel	3.0
RMKDEVP	iRMK Kernel PL/M-386 ASM386 80386 Utilities	2.0
RMKDEVC	iRMK Kernel iC-386 ASM386 80386 Utilities	2.0

SSC 500
Applies to the XENIX Operating System

Product	Description		Version
XNX286	INTEL XENIX 286		3.5 UP2
XNX	INTEL XENIX 386		3.5 UP2
XNX2865K	MS BASIC Interpreter		5.4
XNX2867K	MF COBOL		2.5
X286ASM286	ASM286		3.2
	ASM286	1.2	
	BND286	3.2	
	BLD286	3.2	
	LIB286	3.2	
	MAP286	3.2	
	OVL286	3.2	
X286C386	iC-386		1.0
X286FOR286	FORTRAN 286		3.1
X286PLM286	PL/M-286		3.1
X286ASM386	ASM386		2.1
X286PLM386	PL/M-386		3.1
X286RLL386	386 R & L TOOLSET		1.1
	BLD386	1.2	
	BND386	1.1	
	LIB386	1.1	
	MAP386	1.1	
S386X	ICE 386		1.1
ISVMFCBLAN	COBOL ANIMATOR		1.0
ISVMFCBLCP	COBOL COMPILER		1.0
ISVMFCBLFM	COBOL FORMS UTILITY		1.0
ISVMFCBLRT	COBOL RUNTIME		1.0
MON386	MON386		1.0

SSC 510
Applies to the iDIS Family of Office Automation Software Products
(Includes XENIX 286 Operating System)

Product	Description		Version
XNX286	INTEL XENIX 286		3.5 UP2
XNX	INTEL XENIX 386		3.5 UP2
XNXIBASE	iDIS BASE		2.0 UP1
XNXIDB	iDB		2.0
XNXIMENU	iMENU		2.0
XNXIWORD	iWORD		2.0
XNXIPLAN	iPLAN		2.0
XNXIBPC	DOSNETVTSKR2.1		2.1
XNXIDESK	iDESK		2.0
XNX2865K	MS BASIC Interpreter		5.4
XNX2867K	MF COBOL		2.5
X286ASM286	ASM286		3.2
	ASM286	1.2	
	BND286	3.2	
	BLD286	3.2	
	LIB286	3.2	
	MAP286	3.2	
	OVL286	3.2	
X286C386	iC-386		1.0
X286FOR286	FORTRAN 286		3.1
X286PLM286	PL/M-286		3.1
X286ASM386	ASM386		2.1
X286PLM386	PL/M-386		3.1
X286RLL386	386 R & L TOOLSET		1.1
	BLD386	1.2	
	BND386	1.1	
	LIB386	1.1	
	MAP386	1.1	
S386	ICE 386		1.1
ISVMFCBLAN	COBOL ANIMATOR		1.0
ISVMFCBLCP	COBOL COMPILER		1.0
ISVMFCBLFM	COBOL FORMS UTILITY		1.0
ISVMFCBLRT	COBOL RUNTIME		1.0
MON386	MON386		1.0

SSC 520

Applies to the iDIS Family of Host Communications Software Products
(Includes XENIX 286 Operating System and Basic iDIS Office Automation Software)

Product	Description		Version
XNX286	XENIX 286		3.5 UP2
XNX	XENIX 386		3.5 UP2
XNXIBASE	iDIS BASE		2.0 UP1
XNXIDB	iDB		2.0
XNXIDESK	iDESK		2.0
XNXIMENU	iMENU		2.0
XNXIWORD	iWORD		2.0
XNXIPLAN	iPLAN		2.0
XNXIBPC	iPC		2.0
HASP188	iHASP		2.0
RBTE188	iRBTE		2.0
3270BSC188	i3270BSC		2.0
VPM188	iVPM		2.1
XNX2865K	MS BASIC Interprete		r 5.4
XNX2867K	MF COBOL		2.5
X286ASM286	ASM 286		3.2
	ASM286	1.2	
	BND286	3.2	
	BLD286	3.2	
	LIB286	3.2	
	MAP286	3.2	
	OVL286	3.2	
X286C386	iC-386		1.0
X286FOR286	FORTRAN 286		3.1
X286PLM286	PL/M-286		3.1
X286ASM386	ASM386		2.1
X286PLM386	PL/M-386		3.1
X286RLL386	386 R & L TOOLSET		1.1
	BLD386	1.2	
	BND386	1.1	
	LIB386	1.1	
	MAP386	1.1	
S386	ICE 386		1.0
ISVMFCBLAN	COBOL ANIMATOR		1.0
ISVMFCBLCP	COBOL COMPILER		1.0
ISVMFCBLFM	COBOL FORMS UTILITY		1.0
ISVMFCBLRT	COBOL RUNTIME		1.0
MON386	MON386		1.0
3274SNA88	13274SNA (386 Systems Only)		2.0
SNA/3270	188/56 SNA		1.0

SSC 600 and 6XX
Applies to Network

Product	Description	Version
iNA 960	ISO TRANSPORT LAYER SOFTWARE	3.0
iNA 961	PRECONFIGURED iNA 960 R2.0	2.0
iNA 961	PRECONFIGURED iNA 960 R1.3	1.3
MAPNET	MAP V2.1 SOFTWARE	1.1
MS NET	MS-DOS NETWORK SOFTWARE	1.1
RMX NET	RMX NETWORK SOFTWARE	3.0
XNX NET	XENIX NETWORK SOFTWARE (including iNA 961 R1.3 and R2.0)	2.1
PCLNK	PC-LINK SOFTWARE(V1.0 only)	1.0
MDX465	OpenNET NRM SOFTWARE	3.3
VMSSVR	VAX VMS NETWORKING SOFTWARE	2.0
MVMSSVR	MICROVAX NETWORKING SOFTWARE	2.0
DOSNETVTS	DOSNET VIRTUAL TERMINAL (iPC)	2.1
XNXNETVTS	XNXNET VIRTUAL TERMINAL	2.1
RMXNETVTS	RMXNET VIRTUAL TERMINAL	3.0
SIVMSNET	SIVMSNET For VAX	2.0
SIMVMSNET	SIVMSNET For MICROVAX	2.0
HRX25SU	X.25	1.0
3274SNA88	i3274SNA	2.0
SNA/3270	188/56 SNA (386 Systems Only)	1.0
HASP188	iHASP	2.0
VPM188	iVPM	2.1
RBTE188	iRBTE	2.0
3270BSC188	i3270BSC	2.0
PCLINK2	PCLINK2 Rel. 3 (w/iNA961 R3.0)	3.0
iPC	iPC (DOSNET VT)	2.1 UP 1
SVNET	System V-OpenNET Release 3.2PP	3.2

SSC D700 and G700
Applies to DOS Environment — All Architectures

Product	Description		Version
D86ASM51	ASM51 SINGLE-USER (16-bit)		2.4
	ASM51	2.3	
	LIB51	1.1	
	RL51	3.1	
	OH	1.1	
D86ASM86	iAPX-86 (ASM, R&L) 4.1NL		
	ASM86	3.2	
	CREF86	1.0	
	LIB86	2.1	
	LINK86	3.0	
	LOC86	2.5	
	OH86	1.0	
D86ASM96	ASM96		1.2NL
	ASM96	1.2	
	LIB96	1.2	
	RL96	2.3	
	FPAL96.LIB	1.2	I
	OH	1.1	
D86ASM286	iAPX 286 (ASM, R&L)		3.2NL
	ASM286	1.3	
	BLD286	3.2	
	BND286	3.2	
	LIB286	3.2	
	MAP286	3.2	
	OVL286	3.2	
D86ASM386	ASM386		4.0
D86C86	iC-86		4.1
D86C286	iC-286 SINGLE-USER		4.1
D86EDI86	AEDIT		2.2NL
D86FOR86	FORTRAN 86		3.1NL
D86PAS86	PASCAL 86		3.4 I
D86PAS286	PASCAL 286		3.1NL
D86PLM86	PL/M-86		3.3
D86PLM96	PL/M-96		1.2NL
	FPAL96.LIB	1.2	I
	LIB96	1.2	
	PL/M-96	1.2	
	RL96	2.3	
	OH	1.1	
D86PLM51	PL/M-51 SINGLE-USER (16-bit) 1.3NL		
	LIB51	1.1	
	PL/M-51	1.3	
	RL51	3.1	
	OH	1.1	
D86PLM286	PL/M-286		3.3
D86C386	iC-386		1.2
D86C96	iC-96 SINGLE-USER		1.1
D86PLM386	PL/M-386 SINGLE-USER		3.3

Product	Description		Version
iEDIT	iEDIT		1.0
iPAT286DOS	iPAT286 DOS		1.2
iPATSADOS	iPAT Standalone DOS		2.0
iPAT86DOS	iPAT86 DOS		1.2
III 904D	8086/8088 PROBE SW		1.9
III 914D	80186/80188 PROBE SW		1.9
III 924D	80286 PROBE SW		1.9
III 954D	HOST SOFTWARE		2.0
D86PSC86	PSCOPE		1.1NL
SBE 96	SBE-96		3.1
VLSICE-96	VLSiCE-96		3.0
SA252	ICE 5100/252		3.0
SA452	ICE 5100/452		3.0
SA044	ICE 5100/044		3.0
D86RLL386	386 R & L TOOLSET		1.3
	BLD386	1.5	
	BND386	1.4	
	LIB386	1.2	
	MAP386	1.2	
	OH386	1.0	
	Ease Of Use Kit	1.1	
S386D	ICE 386 - 16 Mhz		1.2
S38620D	ICE 386 - 20 Mhz		1.2
SWICE186	ICE-186		3.0
SWICE286	ICE-286		2.1
D86FOR286	FORTRAN 28		6 3.1NL
86FOR386	FORTRAN 386		4.0
PPS	iPPS for iUP200A/201A		2.3
PPS	iPPS for PCPP		2.3
PCPPSW	PCPP/iPPC		2.0
MON386	MON386		1.0
IPLSII	LOGIC PROGRAMMING SOFTWAR		E 2.0
C960D	iC960 C Compiler		1.2
C960E	eC960 C Compiler		1.2
ASM960D	80960 Assembler & Utilities		1.2
	ASM960	1.2	
	ARC960	1.2	
	DIS960	1.2	
	DMP960	1.2	
	LNK960	1.2	
	MPP960	1.2	
	NAM960	1.2	
	ROM960	1.2	
	SIZ960	1.2	
	STR960	1.2	
EVA960KB	80960 Execution Vehicle & SDM		1.1
K386SX	386SX Probe		1.0
K376SX	376SX Probe		1.0
K386/25	386/25 Probe		1.0

SSC D51 and G51
Applies to DOS 51 Architecture

Product	Description		Version
D86ASM51	ASM51 SINGLEUSER (16-bit)		2.4
	ASM51	2.3	
	LIB51	1.1	
	RL51	3.1	
	OH	1.1	
D86EDI86	AEDIT		2.2NL
D86PLM51	PL/M-51 SINGLEUSER (16-bit)		1.3NL
	LIB51	1.1	
	PL/M-51	1.3	
	RL51	3.1	
	OH	1.1	
SA252	ICE 5100/252		3.0
SA452	ICE 5100/452		3.0
SA044	ICE 5100/044		3.0

SSC D96 and G96
Applies to DOS 96 Architecture

Product	Description		Version
D86ASM96	ASM96		1.2
	ASM96	1.2	
	LIB96	1.2	
	RL96	2.3	
	FPAL96	2.2	
	OH	1.1	
D86EDI86	AEDIT		2.2NL
D86PLM96	PL/M-96		1.2
	FPAL96.LIB	1.2	
	LIB96	1.2	
	PL/M-96	1.2	
	RL96	2.3	
	OH	1.1	
D86C96	iC-96 SINGLEUSER		1.1
SBE 96	SBE-96		3.1
VLSICE-96	VLSiCE-96		3.0
SA196	ICE-196 PC		2.0
ICE196HX	ICE196HX		1.0

SSC D86 and G86
Applies to DOS 86/186 Architecture

Product	Description		Version
D86ASM86	iAPX-86 (ASM, R&L)		4.1NL
	ASM86	3.2	
	CREF86	1.0	
	LIB86	2.1	
	LINK86	3.0	
	LOC86	2.5	
	OH86	1.0	
	DB86	1.0	I
D86C86	iC-86 (Intel C)		4.1
D86EDI86	AEDIT		2.2NL
D86FOR86	FORTRAN 86		3.1
D86PAS86	PASCAL 86		3.4 I
D86PLM86	PL/M-86		3.3
iPATSADOS	iPAT Standalone DOS		2.0
SWICE186	ICE-186		3.0 I
D86PSC86	PSCOPE		1.1NL
III 904D	8086/8088 PROBE SW		1.9
III 914D	80186/80188 PROBE SW		1.9
iPAT86DOS	iPAT86 DOS		1.2
iEDIT	iEDIT		1.0

SSC D286 and G286
Applies to DOS 286 Architecture

Product	Description		Version
D86ASM286	iAPX 286 (ASM, R&L)		3.2NL
	ASM286	1.3	
	BLD286	3.2	
	BND286	3.2	
	LIB286	3.2	
	MAP286	3.2	
	OVL286	3.2	
D86C286	iC-286 SINGLEUSER		4.1
D86EDI86	AEDIT		2.2NL
D86PAS286	PASCAL 286		3.1NL
D86PLM286	PL/M-286		3.3
iEDIT	iEDIT		1.0
iPATSADOS	iPAT Standalone DOS		2.0
UPICE286	ICE-286		2.1
III 924D	80286 PROBE SW		1.9
III 954D	HOST SOFTWARE		2.0
D86FOR286	FORTRAN 286		3.1NL
iPAT286DOS	iPAT286 DOS		1.2

SSC D386 and G386
Applies to DOS 386 Architecture

Product	Description		Version
D86ASM386	ASM386		4.0
D86EDI86	AEDIT		2.2NL
D86C386	iC-386		1.2
D86PLM386	PL/M-386 SINGLEUSER		3.3
D86RLL386	386 R & L TOOLSET		1.1NL
	BLD386	1.5	
	BND386	1.4	
	LIB386	1.2	
	MAP386	1.2	
	OH386	1.0	
	Ease Of Use Kit 1.1		
S386D	ICE 386		1.2
MON386	MON386		1.0
D86FOR386	FORTRAN 386		4.0

SSC 800
Applies to Ada386 Environment

Product	Description	Version
VVSADA386R1.1	386 ADA On Microvax	R1.1
MVVSADA386R1.1	386 ADA On VAX	R1.1
VVSADARMKR1.1	RMK ADA On Microvax	R1.1
MVVSADARMKR1.1	RMK ADA On VAX	R1.1
U386ADA386R1.0	386 ADA On UNIX	R1.0

SSC D960 and G960
Applies to DOS 960 Architecture

Product	Description		Version
D86EDI86	AEDIT Text Editor		2.2NL
C960D	iC960 C Compiler		1.2
C960E	eC960 C Compiler		1.2
ASM960D	80960 Assembler & Utilities		1.2
	ASM960	1.2	
	ARC960	1.2	
	COF960	1.2	
	DIS960	1.2	
	DMP960	1.2	
	LNK960	1.2	
	MPP960	1.2	
	NAM960	1.2	
	ROM960	1.2	
	SIZ960	1.2	
	STR960	1.2	
EVA960KB	80960 Execution Vehicle & SDM		1.1
ICE960KB	Incircuit Emulator		1.1

SSC U960
Applies to Intel's SUN3 Sun O.S. V3.2/V3.5 Resident Cross-Software

Product	Description		Version
C960U	iC960 C Compiler		1.2
ASM960U	80960 Assembler & Utilities		1.2
	ASM960	1.2	
	ARC960	1.2	
	COF960	1.2	
	DIS960	1.2	
	DMP960	1.2	
	LNK960	1.2	
	MPP960	1.2	
	NAM960	1.2	
	ROM960	1.2	
	SIZ960	1.2	
	STR960	1.2	
EVA960KB	80960 Execution Vehicle & SDM		1.1

SSC 1000
Applies to Logic Programming

Product	Description	Version
iPLSII	Logic Programming Software	2.0

REQUEST FOR READER'S COMMENTS

Intel's Customer Support Operation publishes **Troubleshooting Guides** for you. Please let us know what you think. We provide this post-paid form for your convenience.

If you tell us something you like, we'll do more of it. If you tell us something you dislike, we'll attempt to improve that area.

Please restrict your comments to the publication—the topics covered, readability, usefulness, and accuracy. If your comment is about the product itself, you should contact your Intel Representative.

ISSUE
(quarter/year)_____

1. Did you come across any technical errors? If so, where? (Please specify the page number.)
_____
_____
_____
_____

2. Did you find the information useful? Is it too elementary? Too advanced?
_____
_____
_____
_____

3. Have you encountered a problem you feel needs investigation? Please describe briefly.
_____
_____
_____
_____

4. Can you find the product of interest easily within this publication?
_____
_____
_____
_____

5. Please rate this publication on a scale of 0 to 4. _____
(0 = Very Poor;   1 = Poor;   2 = Good;   3 = Very Good;   4 = Excellent)
You can single out individual chapters. If you wish you may rate each chapter individually.
_____
_____
_____

NAME _____ DATE _____

TITLE _____

COMPANY NAME/DEPARTMENT _____

ADDRESS _____

CITY_____ STATE _____ ZIP CODE _____
(COUNTY)

WE'D LIKE YOUR COMMENTS . . .

Your comments on the back of this form will help us provide you with better support. Each reply will be carefully reviewed by the responsible person. All comments and suggestions become the property of Intel Corporation.

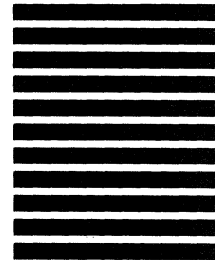_____ _____ _____ _____ _____ _____ FOLD HERE _____ _____ _____ _____ _____ _____

_____ _____ _____ _____ _____ _____ FOLD HERE _____ _____ _____ _____ _____ _____