

Convergent Technologies

ADDENDUM NOTICE

for

6.2-2 CTIX Addendum

(SAC-20Bx, SAC-2014x, SAC-20Cx, SAC-20Fx)

Revised August 15, 1990

Trademark Notice

Convergent Technologies, NGEN, MegaFrame and MightyFrame are registered trademarks of Convergent Technologies, Inc.

WGS/Office, WGS/Desktop Manager, WGS/Word Processor, WGS/Spreadsheet, WGS/Mail, WGS/Calendar, Workgroup Solutions, PC Exchange, S/640, S/480, S/320, S/222, S/221, S/280, S/220, S/120, S/80, MiniFrame, AWS, IWS, Server PC, PT, GT, CTIX, CTIX/386 and CTOS are trademarks of Convergent Technologies, Inc.

CTIX and CTIX/386 are derived from UNIX System V software, under license from AT&T. UNIX is a trademark of AT&T.

·
·
C

C

C

TABLE OF CONTENTS

SECTION	TITLE	PAGE
1.	Description of the Addendum	1
1.1	Contents of the Software Package	1
1.2	General Overview of the Addendum	1
2.	Changes from the Current Release	3
2.1	Kernel Support for the IOP16 Expansion Board	3
2.2	Commands Support for the IOP16 Expansion Board	3
2.3	Adman Support for the IOP16 Expansion Board	3
2.4	Change to the Kernel Notify Routines	4
2.5	CROSS Group Installation	4
2.6	Compatibility Module S for z8530.c	4
2.7	SCSI Disk Performance Improvements using I/O Collapsing	6
2.8	Configuring the System for the 4K Block File System	7
2.9	Improved SCSI Error Recovery	7
2.10	SCSI Controller Number Assignment	8
2.11	Support for the Half-height Quantum and Micropolis Drives	9
2.12	Adman and 3.00NS CTAM Enhancements	9
2.13	New STREAMS Module and uugetty(1M)	10
2.14	Additional System Files	11
2.15	External SPR's Fixed in this Addendum	11
2.16	Internal SPR's Fixed in this Addendum	12
3.	Contents of the CTIX Addendum Media	14
3.1	CTIX Addendum Tape	14
4.	CTIX Addendum Installation Requirements	17
4.1	Files for a CTIX Addendum Installation	17

4.2	Space for an Addendum Installation	17
5.	Installation Procedure	19
5.1	Addendum Installation	19
5.2	Addendum Installation using Silent Update	23
6.	Known Errors, Warnings, and Restrictions	24
6.1	Known Errors	24
6.2	Warnings	24
7.	Documentation Updates	30
	Appendix A: Port Selection	31

1. Description of the Addendum

1.1 Contents of the Software Package

The **6.2-2 CTIX Addendum** ordered by **SAC-20Bx,SAC-2014x,SAC-20Cx,SAC-20Fx** contains:

- 6.2-2 CTIX Addendum Notice
(Part Number B-09-02414-B)
- 6.2-2 Documentation Update
(Part Number B-09-01977-01-F)
- 6.2-2 CTIX Addendum Tape
(Part Number 71-03495-01)

The 6.2-2 CTIX Addendum software release is on one tape.

1.2 General Overview of the Addendum

This is an enhancement update to the previous release of CTIX. It is applicable either to the Development or the Runtime version of 6.2 CTIX. It contains:

1. Support for the IOP16 Expansion Board.
2. Fixes for some STREAMS tty problems.
3. Fixes affecting IOP Accelerator Board performance.
4. Compatibility module S and SX for Z8530.c.
5. Performance improvements on SCSI disk 4K Filesystem.
6. Improved SCSI error recovery (Archive Timeout).
7. Support for the new Half-Height Quantum 168MB and Micropolis 340MB disk drives.
8. New 3.00NS CTAM standardized keyboard map, font, and terminfo files, along with 3.00NS CTAM support for the TO-300 PC Keyboard.
9. There is a new STREAMS module, **mdm**, and a new version of *uugetty(1M)* to handle open with delay vs open with no delay.

10. Fixes to some commands and utilities.

Please note that the Addendum tape is not a bootable tape. If the installation is interrupted, DO NOT use /install/restart! Simply redo the steps detailed in the Installation Section.

2. Changes from the Current Release

The content of the CTIX Addendum is described below.

2.1 Kernel Support for the IOP16 Expansion Board

1. The host IOP driver, **tiop**, has been enhanced to support the IOP16.
2. There is a new downloadable image, **iop16**, for the IOP16 Expansion Board.
3. The auto-configuration script **/etc/drvload** is changed to include the loading of **/etc/lldrv/iop16**.

2.2 Commands Support for the IOP16 Expansion Board

There are changes to some commands to support IOP16 boards.

1. **serstat(1M)** and **hinv(1M)** now support the IOP16. There are updated man pages in the *Documentation Update*.
2. **crash(1M)** kernel **tty** structure reporting has been fixed and enhanced to support the IOP16. A new command has been added: **iop16**. (or **tty -t iop16**). There is an updated man page in the *Documentation Update*.
3. There is a new utility **iopdump(1M)** that uploads the contents of memory in an IOP or IOP16 and prints the image in binary or hexadecimal characters to standard out. There is a new man page in the *Documentation Update*.

2.3 Adman Support for the IOP16 Expansion Board

Necessary changes were made so that the IOP16 Expansion Board is fully supported by **adman(1)**. The changed files are:

1. **/usr/lib/adman/getboards**
2. **/usr/lib/adman/getchanls**
3. **/usr/lib/adman/english_usa/term_msg.rf**

2.4 Change to the Kernel Notify Routines

The kernel *notify* routines have been modified to run in a STREAMS environment.

2.5 CROSS Group Installation

The installation of the CROSS group has been improved. A new version of */cross/crossins* is provided for reinstallation of CROSS, if necessary. Also fresh copies of */etc/profile* and */etc/cprofile* are provided.

If you were not able to successfully install the CROSS group, the CROSS group files in this Addendum will assist you in reinstallation. If you want to reinstall CROSS, do the following:

1. Bring your system to single-user level, as described in Section 5.1.
2. If you have CROSS libraries in a file system mounted on */cross*, mount the file system. Remove the files in */cross*:

```
rm -rf /cross/*
```

3. Insert the 6.2 CTIX Installation Tape in your system and install the CROSS group alone, using the **silent** option.
4. Insert the 6.2-2 CTIX Installation Tape in your system and install the CROSS group, using **silent** option.
5. Execute:

```
/cross/crossins
```

You are then prompted, in clear terms, for your choices for installation.

2.6 Compatibility Module S for z8530.c

The following routines are provided in 6.2-2 CTIX and later versions of CTIX for compatibility among different machines. These are included in the **z8530.o** module which is standard in

all releases. The **aZ*** routines are provided since the base addresses and offsets of the z8530 ports are different between S/Series machines. These routines return the address of the structure members (**cmd**, **data**, **exstatus**, **extclk**). A return of zero is given for non-existing ports. The communication software uses only the two base z8530's, but these routines are valid for any existing z8530 port.

Specific details of the routines are given below. The first three routines already exist in **compat** and have not been modified. The last four routines are new but currently unused features of **compat**. They will be used by future versions of the Data Communications products.

1. **int plug_svec(drvid, dev, rx, tx, sr, ex)**
Common routine to plug in z8530 device interrupt routines. Returns:
 - < 0 if in use (by some other driver)
 - = 0 if OK and caller owns it now
 - > 0 if caller already owned it

2. **unplug_svec(drvid, dev)**
Common routine to free z8530 device interrupt routines.

3. **int dsrstat(dev)**
Returns non-zero if and only if DSR input status is active.

4. **caddr_t aZcmd(dev)**
Returns address of the command register for a z8530 port.

5. **caddr_t aZdata(dev)**
Returns address of the data register for a z8530 port.

6. **caddr_t aZexstatus(dev)**
Returns address of the external status register for a z8530 port.

7. **caddr_t aZextclk(dev)**
Returns address of the external clock register for a z8530 port.

2.7 SCSI Disk Performance Improvements using I/O Collapsing

SCSI Disk I/O Collapsing tries to combine multiple SCSI disk I/O requests into a single request if those requests try to read or write consecutive disk blocks. Multiple requests are collapsed into one to reduce the overhead of the SCSI driver and SCSI bus protocol and to increase the SCSI driver performance substantially.

Although the data blocks on disk are consecutive, the buffers in the kernel are not likely to be. To allow the DMA controller to transfer data into or out of non-contiguous buffers, the virtual addresses of the buffers are remapped into contiguous kernel I/O map addresses. Since the I/O page size on S/Series systems is 4K, this feature can only work for I/O requests which transfer data in multiples of 4K (4K, 8K, 12K...). The 1K file system does not use this feature.

There are also four new Tunable Parameters. They are:

nra, **nra4k**, **v_hbuf**, and **v_h4kbuf**.

nra is the number of records to look ahead for the 1K file system (default == 4).

nra4k is the number of records to look ahead for the 4K file system (default == 16).

v_hbuf is the number of buffer cache hash buckets for the 1K file system. If the user does not specify one, CTIX automatically configures using the following formula:

$$v_hbuf = v_buf(\text{number of 1K buffers})/2$$

then round up to the next higher number that is a power of 2.

For example:

if $v_buf = 1024$, then $v_hbuf = 512$

if $v_buf = 512$, then $v_hbuf = 256$

if `v_buf = 384`, then `v_hbuf=256`

`v_h4kbuf` is the `v_hbuf` version for 4K file system.

The user can change the parameters `nra` and `nra4k` while CTIX is running. For the parameters `v_hbuf` and `v_h4kbuf`, the user must list the new values in the file `/etc/system`, execute `uconf -w`, and reboot.

2.8 Configuring the System for the 4K Block File System

If you want to create 4K file systems, you must configure your system to optimize performance.

1. The system should have at least 3MB of physical memory.
2. The 4K file systems should be created on disks of size 85MB or more.
3. The number of 4K buffers should be about one-fourth the number of 1K buffers. The number of 1K buffers is auto-configured at bootup. Obtain the number of 1K buffers using `uconf -v v_hbuf`.

Let that value be called `val1k`. Calculate the required number of 4K buffers:

$$\text{val4k} = \text{val1k} / 4$$

Configure the number of 4K buffers accordingly using `uconf v_h4kbuf=val4k`.

2.9 Improved SCSI Error Recovery

There are several changes to the SCSI driver regarding the way it handles error conditions. In previous versions of the driver, the initiator disconnected whenever it encountered a condition it regarded as *not normal*. This resulted in SCSI bus hangs as the target waited for the initiator to respond. Because SCSI is a target-driven system, the initiator must wait for the target to specify its intentions before the former decides what to do.

This version of the driver does not make as many unilateral decisions for certain conditions, informs the target that the operation needs to be aborted, and waits for the target to

disconnect before telling the initiator to do so.

2.10 SCSI Controller Number Assignment

The assignment of controller numbers needs some clarification. Determining the default controller numbers for disks is straight-forward, but determining the numbers for tapes is not as simple.

The S/Series systems can be divided into three groups: (A) S/120,22x,320; (B) S/480,640; (C) S/80,280. Each group is discussed separately. The disks and tapes are divided into boot disk, user disk, boot tape, VME Half-Inch Tape, and SCSI tape.

Group A: S/120,22x,320

Device	Bus Type	Dev Node
-----	-----	-----
Boot disk	ST506	/dev/dsk/c0d0
SCSI disk	SCSI	/dev/dsk/c2dx
Boot tape	QIC2	/dev/rmt/c0d0
VME HIT	VME	/dev/rmt/c1dx
SCSI tape	SCSI	/dev/rmt/c0d[1-7]

Group B: S/480,640

Device	Bus Type	Dev Node
-----	-----	-----
Boot disk	SCSI	/dev/dsk/c0d0
SCSI disk	SCSI	/dev/dsk/c0d[1-3]
ST506 disk	ST506	/dev/dsk/c2dx
Boot tape	SCSI	/dev/rmt/c0d0
VME HIT	VME	/dev/rmt/c1dx
SCSI tape	SCSI	/dev/rmt/c0d[1-7]

Group C: S/80,280

Device	Bus Type	Dev Node
-----	-----	-----
Boot disk	SCSI	/dev/dsk/c0d0
SCSI disk	SCSI	/dev/dsk/c0d[1-3]
Boot tape	SCSI	/dev/rmt/c0d0

2.11 Support for the Half-height Quantum and Micropolis Drives

The Quantum 168Mb and Micropolis 340MB Half-height drives are supported. The description files for the drives already existed on the 6.2-1 CTIX Addendum tape.

WARNING: The description file for Quantum 170S is incorrectly named desc.120S.

2.12 Adman and 3.00NS CTAM Enhancements

The TO-300 PC keyboard is supported.

New files are introduced to enhance the number of terminals supported by **3.00NS CTAM** (non-STREAMS version). These are the kmap files. Kmap files describe the functional mapping of keys on the terminal. Generally, there are two classes of keys on the keyboard; there are functional keys, which give direct access to a function via a single keystroke, and control keys, which give access to a function by pressing multiple keys. These new files define a common set of control keys that provide the same functionality across all terminal types.

In addition, the currently supported kmap files are modified to also define a common set of control keys. See below for those terminals not supported.

The changes can be described by example. For the CTIX administrative tool, *adman(1)*, **<CONTROL>J** now issues the same function for **all** supported terminals. If a particular terminal has a **<DO>** or **<Enter>** key on the keyboard, then these functional keys are the displayed key. The **<CONTROL>** key also works, and for keyboards that do not have the function keys builtin, the control sequences appear as the default.

The control keys have been moved around; therefore, if an application assumes certain control keys have certain functions, modifications might be required for the new mappings.

Files that have been modified:

**ct235.kb, ct250.kb, ct300.kb, pt.kb, to250.kb, to300.kb,
vt100.kb, vt220.kb, wy85.kb, ct235.ft - add smacs3**

New keyboard map files:

**pcu.kb, pcucolor.kb, uvt101.kb, uvt1224.kb,
uvt1224g.kb, uvt1224gpc.kb, to101.kb**

New font files:

**pcu.ft, pcucolor.ft, uvt101.ft, uvt1224.ft,
uvt1224g.ft, uvt1224gpc.ft, to101.ft**

New terminfo files:

**pcu.ti,
pcucolor.ti, uvt101.ti, uvt1224.ti, uvt1224g.ti,
uvt1224gpc.ti, att.ti**

No *terminfo* files are being modified, only added.

Keyboard maps and fonts files for the following terminals did not change, because 3.00NS CTAM no longer supports them:

bt970, fortune, l220 (link 220), tvs9220.

Therefore, the consistent interface is applicable only to those terminals supported by 3.00NS CTAM and not necessarily those supported by CTIX.

2.13 New STREAMS Module and *uugetty(1M)*

uugetty(1M) had a problem with the STREAMS single-threaded **open** characteristic. The new approach is to have *uugetty(1M)* open the tty port with **O_NDELAY**, to get around the single-threaded open that STREAMS imposes. *uugetty(1M)* pushes a new STREAMS module, **mdm**, that provides STREAMS messages when **CD** or **CTS** status lines change, and calls

getmsg(2) to get these messages. This allows having a routine that can wait (block) for carrier to become present without polling.

The new *uugetty(1M)* handles non-STREAMS tty ports like the RIOP, and also works if it cannot push the *mdm* module on the stream. In the first case, it just issues an open without **O_NDELAY**; in the second, it uses **TCGEXT** to poll for **CD** every 5 seconds.

2.14 Additional System Files

The library, */usr/lib/libcrash.a*, omitted from the 6.2 CTIX release, is included in this Addendum.

2.15 External SPR's Fixed in this Addendum

1. (TS SPR #16119) A kernel panic in *uts/common/os/streamio.c* (line 305) is fixed.
2. (TS SPR # 16086) The installation of the CROSS group has been improved. A new version of *crossins* is provided for user reinstallation, if necessary.
3. (TS SPR #16084 and 16117) The kernel parameters **v_hbuf** and **reccs** have been removed from *uconf(1M)*. These parameters must be changed by doing a kernel build.
4. (TS SPR #15501) A bug in the kernel that caused an RFS client cache not to be updated is fixed.
5. (TS SPR #15333) IOP ports now correctly handle **CARRIER** when opened.
6. (TS SPR #16187) *cu(1)* does not disconnect when **CARRIER** drops as it used to.
7. (TS SPR #16137) *uucp(1C)* using callback doesn't work.
8. (TS SPR #16095) */usr/sys/cf/Makefile* has no dependency from *dfile* to **conf.o** and **low.o**.
9. (TS SPR #16097) Broken escape sequences on IOP's.

10. (TS SPR #16244) A serial printer on a bare port using the print spooler may cause the port to hang when it runs out of paper.
11. (TS SPR #15694) The Enet driver returns the wrong mblk types for some LLI primitives.

2.16 Internal SPR's Fixed in this Addendum

1. A problem with the z8530 driver is fixed. The `lp` system would get stuck while trying to close a port. The port was hung in `serwait()` which was sleeping, waiting for the write queue to drain.
2. If a serial interface printer has just issued an `XOFF` and then goes offline, and you then kill the print job using `cancel(1)`, the port hangs. The utility `rsterm(1M)` cannot currently recover the port. The only workaround short of rebooting is to force an `XON` from the printer after it is brought back online.
3. The utility `crash(1M)` does not report correct line discipline information for STREAMS-based tty ports. The new STREAMS version of the IOP and IOP16 kernel driver, `/etc/lldrv/tiop`, does not currently update part of the traditional `tty structure`, which `crash(1M)` uses and `stty(1)` does not. If no echo, `~echo`, is set using `stty(1)`, `crash(1M)` incorrectly reports that echo is on.
4. If a port is closed while it is in the `tblk` state (flow control is off), the blocked state is not cleared when the port is closed. This condition might occur if for example a serial device sends an `XOFF` and then goes off-line. Use `rsterm(1M)` to clear the blocked state.
5. `cron(1M)` now correctly runs jobs for all users in `/usr/spool/cron/crontabs`.
6. All ports controlled by an IOP now correctly lower `DTR`.
7. IOP ports no longer report underruns that clog `/etc/log/confile` and force the IOP into saturation of the bus.
8. The `CHECK_MODEM` request now returns the status of the `CARRIER` signal.

9. Bare ports no longer get stuck in the *close()* routine.
10. *crash(1M)* now correctly reports kernel tty structures for clist-based I/O ports.
11. Tty ports now correctly send **SIGHUP** when **CARRIER** is dropped.
12. An Ethernet driver bug in the kernel that caused buffers to be lost is fixed.
13. *termio(7)* **O_NLCR** now works properly.
14. A bug in the debugger routines in the kernel that caused the debugger to hang has been fixed.
15. A STREAMS bug that broke type-ahead in *vi(1)* has been fixed.
16. The linekill character is now correctly handled for STREAMS tty.
17. *crash(1M)* does not report the correct STREAMS character count.
18. An S/80 SCSI panic **OPERATION TIMEOUT** is fixed.
19. Failure of a system to boot with certain configurations of expansion boards is fixed.
20. Serial ports with **NDELAY** set no longer block.
21. Execution of **scsimap -u** now correctly flushes the disk entries from a previous update from **/etc/system**. Also an invalid entry in the system file can no longer cause the kernel **scsimap** to become corrupt.
22. *createdev(1M)* does not correctly create **tty** device nodes with port number **256** or greater.
23. *iopdump(1M)* only obtains the first IOP16 board's memory.
24. A new STREAMS module **mdm** is added to fix a *uugetty(1M)* problem with **O_NDELAY** and single threaded STREAMS interrupts.

3. Contents of the CTIX Addendum Media

3.1 CTIX Addendum Tape

The CTIX Addendum Tape contains 33 tape files. Files 5 through 32 are `cpio` archives in `-cQ` format.

<i>File #</i>	<i>Description</i>
0	VHB and loader
1	(reserved)
2	(reserved)
3	(reserved)
4	(reserved)
5	installation tools
6	installation control files for ADDEN group
7	customizable files for ADDEN group
8	required files for ADDEN group
9	installation control files for MISC group
10	required files for MISC group
11	installation control files for ADMAN group
12	required files for ADMAN group
13	installation control files for LP group
14	required files for LP group
15	installation control files for UUCP group
16	required files for UUCP group
17	installation control files for LAN group
18	required files for LAN group
19	installation control files for SYSM group
20	required files for SYSM group
21	installation control files for TINFO group
22	required files for TINFO group
23	installation control files for RIOP group
24	required files for RIOP group
25	installation control files for IOP16 group
26	required files for IOP16 group
27	installation control files for BCE group
28	required files for BCE group
29	installation control files for CROSS group
30	required files for CROSS group
31	installation control files for KFIG group

This release is structured to allow installation of one or more selected subsystems, or groups of commands. A brief description of each of the groups follows.

<i>Group</i>	<i>Description</i>
ADDEN	Required CTIX ADDEN commands. This group is mandatory. You must install this group.
MISC	MISCellaneous CTIX commands.
ADMAN	User administration via the <i>adman</i> menu driven program. This group contains changes to support the IOP16 board.
LP	Line Printer spooling system.
UUCP	UNIX system to UNIX system CoPy programs.
LAN	Local Area Network protocol independent programs.
SYSM	SYSTEM Maintenance commands including system profiling and file system repair.
TINFO	Terminal INFOrmation files in <i>/usr/lib/terminfo</i> .
RIOP	Remote IOP system.
IOP16	IOP16 required device nodes and driver.
BCE	Basic 'C' Environment, including the 'C' compiler, loader, assembler, libraries. This suite is configured to generate code for a 68020 CPU with software floating point.
CROSS	CROSS Development files. This group contains fresh copies of <i>/etc/profile</i> and <i>/etc/cprofile</i> and an improved version of <i>/cross/crossins</i> .
KFIG	Kernel conFIGuration files and commands. This group contains libraries that are needed to build a new CTIX Kernel. It reflects changes made for the IOP16 drivers and interfaces and for fixes to kernel bugs.

NOTE: The last groups on the Addendum tape, CROSS and KFIG, are not useful in a Runtime environment. If installed, they will take up **2794** blocks of disk space. Refer to the appropriate instructions in the sections on installation.

To obtain an actual list of the contents of the **cpio** archives on the CTIX Addendum Installation Tape, execute:

```
tsioctl -c rewind /dev/rmt0
tsioctl -c skip /dev/rmt4 5
while true
do
    /bin/cpio -icvtQ < /dev/rmt4
    if [ $? = 2 ]
    then
        break
    fi
done
```

4. CTIX Addendum Installation Requirements

4.1 Files for a CTIX Addendum Installation

The following files are required for an addendum installation. If they do not exist on the currently running CTIX system, the installation will fail.

/bin/basename	/bin/cat
/bin/chmod	/bin/cp
/bin/cpio	/bin/df
/bin/dirname	/bin/echo
/bin/expr	/bin/grep
/bin/ln	/bin/ls
/bin/mkdir	/bin/mv
/bin/pwd	/bin/rm
/bin/sh	/bin/sync
/bin/uname	/bin/who
/etc/devnm	/etc/fsck
/etc/hinv	
/etc/rc	/etc/mount
/etc/setmnt	/etc/umount
/usr/local/bin/fsiz	/usr/local/bin/tsioctl
/etc/init	(ADDEN only)
/etc/labelit	(ADDEN only)
/usr/bin/cut	(ADDEN only)

4.2 Space for an Addendum Installation

Installation of the ADDEN group requires approximately **2500** free blocks on the root file system at the time of installation script invocation. The command

```
df /
```

reports the number of free blocks on the root file system. If there are not **2500** free blocks, clean up the file system to obtain them before executing the installation script. This space is required to save the files listed above and to read in the customizable files.

Installing without adequate free space may result in a system with no /unix, which means that rebooting the system will fail.

The following table shows the additional blocks required for each group, and for the entire system, broken down by those which are used under / and /usr.

Group	/	/usr	total
Install	286	0	286
ADDEN	2240	252	2492
MISC	8	0	8
ADMAN	0	856	856
LP	0	74	74
UUCP	0	296	296
LAN	380	0	380
SYSM	396	0	396
TINFO	0	50	50
RIOP	66	0	66
IOP16	40	0	40
BCE	0	364	364
CROSS	8	0	8
KFIG	0	2422	2422
TOTAL	3424	4314	7738

Refer to the appropriate instructions in the sections on installation.

5. Installation Procedure

It is necessary that the current release of 6.2 CTIX be installed on your system disk prior to beginning the addendum installation.

If the installation is interrupted, DO NOT use `/install/restart!` Simply redo the steps detailed in the Installation Section.

If you have gotten your system into a strange, unbootable state, use the 6.2 CTIX Installation tape and install **only** the ADDEN group. After restoring the ADDEN group you must restore various products (data communication packages, etc) that use `/usr/sys` files, removed during the ADDEN installation.

5.1 Addendum Installation

To install the 6.2-2 CTIX Addendum, use `ctinstall(1)` with the `update` or `silent` option, and perform the following steps:

1. Log in as root. Provide a password if necessary. The system responds with a `#` prompt and leaves you in the root directory `/`.
2. Bring the system into single-user mode by entering:

```
/etc/shutdown
```

Before installing the product, wait for the system to prompt with the message:

```
ok to stop or reset processor
```

3. Re-mount `/usr` if it is a mounted file system:

```
mount /usr
```

4. Insert the distribution quarter-inch tape into the drive.
5. Rewind the tape with the following command:

```
tsioctl -c rewind /dev/rmt0
```

6. Execute the installation script:

ctinstall

7. The following output appears on the screen. User supplied responses are shown in **boldface** type; a carriage return is implied after every user input.

```
ctinstall.sh 6.32
```

Positioning the Tape for Product Installation

Update, silent update or new installation of CTIX 6.2-2
('update' 'silent' or 'install')?: **update** or **silent**

8. Your choices are:

```
ADDEN MISC ADMAN LP UUCP LAN SYSM  
TINFO RIOP IOP16 BCE CROSS KFIG
```

If you'd like to install all of the groups, type 'all';
If you'd like some groups, type groups separated by blanks;
If you'd like omit some groups, type 'omit' followed by
groups to omit;
If you'd like none of the groups, type 'none':

Enter the names of the groups you want to install. In this case you should enter **all** if the system is a Development System, or for Runtime use the **omit** option and specify **BCE CROSS KFIG** if it is not.

If you had difficulty with the CROSS group installation of 6.2 CTIX, install **CROSS** and follow the instructions for recovery mentioned in a previous section.

NOTE for the Runtime only system: Do not install the CROSS and KFIG groups. These groups are of no use to you, and will only take up space on your disk.

9. The installation continues by installing the following CTIX 6.2-2 groups on your system into /:

Group(s) ADDEN MISC ADMAN LP UUCP LAN
SYSM TINFO RIOP IOP16 BCE CROSS KFIG

You see messages similar to the following during the installation process:

Starting to Install Group(s) ADDEN MISC ADMAN LP
UUCP LAN SYSM TINFO RIOP IOP16 BCE CROSS
KFIG

Installing Group ADDEN.

Calculating size required for group ADDEN.

934 additional 512 byte blocks will be used on /
1 additional inodes will be used on /
2 additional 512 byte blocks will be used on /usr

Installing Customizable ADDEN files.

Installing required ADDEN files.

Checking permissions and modes on new ADDEN
commands.

Executing ADDEN Install script.

/unix linked to CTIX6.2-2m1 or CTIX6.2-2m2
Completed Installation of Group ADDEN.

Continuing with MISC ...

Installing Group MISC.

Calculating size required for group MISC.

Installing required MISC files.

Checking permissions and modes on new MISC
commands.

Completed Installation of Group MISC.

Continuing with UUCP ...

[And so on for the rest of the groups. This step will be repeated for each Group you have selected.]

Rewinding tape.

Installation Complete.

10. Remove the tape. To configure the kernel, execute:

```
uconf -w
```

Reboot the system to install the new IOP16 drivers and the new CTIX Kernel. Unmount the `/usr` file system by typing:

```
umount /usr
```

Then enter these commands:

```
sync; sync; sync
```

11. If you need to install IOP16 boards enter:

```
reboot -h
```

Turn off the power and install the IOP16 board. Then turn the power on and the system will boot.

12. If you already have the IOP16 boards installed in your system type:

```
reboot
```

Your system reboots.

13. See *Appendix A: Port Selection* for more information on port numbering with respect to IOP16 slot number.

5.2 Addendum Installation using Silent Update

To non-interactively install the addendum on your system, follow these instructions.

1. Perform steps 1 through 5 listed in Section 5.1, "Addendum Installation using Update", above.
2. Execute the installation script:

```
ctinstall silent [GROUPS]
```

For Development environment, 'GROUPS' can be any or all of:

```
ADDEN MISC ADMAN LP UUCP LAN SYM  
TINFO RIOP IOP16 BCE CROSS KFIG
```

and for Runtime environment, use the **omit** option and specify **BCE CROSS KFIG**.

The groups must be specified in the above order, uppercase, and separated by blanks. No questions are asked by the installation script.

6. Known Errors, Warnings, and Restrictions

6.1 Known Errors

All known errors and restrictions of 6.2 CTIX that are not fixed in this Addendum also apply to this version of CTIX. Refer to the 6.2 S/120,S/22X,S/320 CTIX Release Notice for details.

1. Any software which attempts to broadcast (e.g., *routed* or *ruhd*) will not work with *slipd*. The broadcasting software constantly tests the port and causes *slipd* to dial out. If this occurs with *routed*, a workaround is to use static routing (typically a default route in */etc/rcopts/ROUTING*) rather than running *routed*.
2. On an S/280, *hinv(1)* always reports two RS-232 expansion boards, even if there is only one. The problem is in the way that the IOP controlled ports are initialized.
3. For the IOP16, the utility *iopdump(1M)* obtains only the data from the first board.
4. The interactive loader, */usr/lib/iv/loader16*, does not work.
5. Using *iv(1M)* with parameters that describe a disk as larger than it actually is go undetected. When the disk is accessed at a block location beyond its physical limit, errors occur or the system might panic.
6. The command *serstat(1M)* reports a *read error* and terminates, if run on a system without IOP or IOP16.
7. If */bin/pwck* finds an error in */etc/passwd*, *pwconv(1M)* processes the file, returns a successful termination status, but */etc/passwd* and */etc/shadow* are truncated at the last good entry.

6.2 Warnings

1. The description file for Quantum 170S is incorrectly named *desc.120S* under */usr/lib/iv*.
2. The F880 Cipher Half-Inch Tape drive when used with the Cipher CSC-100 SCSI Adapter is not fully functional. The *tapeset(1M)* command does not work correctly. Also early versions of the SCSI Adapter have a problem with **MODE SELECT** for variable density operation.

3. When running *slipd* with options:

slipd is normally run by creating the empty file `/etc/rcopts/SLIPD`. This causes *slipd* to be run with no options at boot time. However, if options are desired (for instance, the `-e` option to deal with slow-connecting modems), the following steps are recommended:

- delete `/etc/rcopts/SLIPD`
- add the following commands to `/etc/rcopts/ROUTING`

```
ODIR = 'pwd'
cd /etc/lldrv
./lldrv -a switch
/etc/slipd [options]
cd $ODIR
```

slipd has a default timeout of 15 seconds. If you are using a slow modem, you might want to use the `-e` option with an argument of 30.

4. Unloading some of the dynamically loadable drivers out of sequence or while the system is in an active state might cause the system to panic. For example, if you are running the printer driver `plp` with *lpsched* running and unload the driver, then reload it and submit some jobs to the spooler, the system panics. Or, if you are running the driver `x25` together with *slipd* and you unload `x25` first, the system panics. To safely unload these drivers you must execute a **shutdown** and change the initialization scripts or flag files so that the driver is not automatically loaded, then reboot.
5. The password aging arguments for `/etc/shadow` are in units of days not weeks. The man page for *shadow(4)* is somewhat unclear about this.
6. *fstyp(1M)* reports any file system (1K or 4K) is of type **S51K**.
7. If your system has an IOP and you execute `errpt -a` and see a large number of messages like *nochars for terminal device 0*, this means that the IOP is running out of memory to store characters. It is an indication that the IOP is overloaded and you must reduce the traffic on

IOP-controlled ports. Note that execution of `netstat -m` does not show any errors.

8. Under exceptionally severe serial port load conditions, it is possible for a system to run out of buffers, which causes tty ports to hang. In most situations, you can still execute commands to detect the problem and recover from the hangs. In the worst case, it might be necessary to reboot the system. This problem can be remedied by increasing the number of allocated buffers (refer to the *6.2 CTIX Release Notice* for further information on buffer allocation) or by using flow control. This problem should not be encountered in normal use.
9. When two adjacent IOP16 ports (0 and 1, 2 and 3, etc.) are used in a loopback configuration and a command like `cat(1)` is used to send data out one port and into the other, one of the ports sometimes hangs waiting for Carrier Detect. This is a result of the hardware implementation of the port.
10. `sleep(3C)` uses `signal(2)`, not `sigset(2)`, to reset the caller's **SIGALRM** handler routine. Therefore the signal action is reset to its default action on execution of the **SIGALRM** handler. This is probably not what the programmer intended if `sigset(2)` had originally been used to set the signal action. `Sleep(3C)` uses a `longjmp`, which returns to the `sleep(3C)` context when the `alarm(2)` signal handler routine is executed. This might cause premature preemption and loss of context from other nested signal handler routines. Refer to the new man page in the *Documentation Update*.
11. Execution of `hinv HW` (where `HW` is one of the hardware component specifiers) returns **0** if the hardware component is present in the system. The man page claims that it returns **1**. Refer to the new man page in the *Documentation Update*.
12. The implementation of `notify(2)` has changed slightly. Refer to the new man page in the *Documentation Update*.
13. If you are using large file systems (above 80 MB), avoid using the `bitmap` version of the file system; refer to the man page for `mkfs(1M)`. The kernel file system services

might take as much as 3 to 5 minutes to update the bitmap during large writes (of 300 to 500 MB).

14. The kernel libraries and **locore.o** in **/usr/sys** are overwritten if the KFIG group is installed. If you are customizing the kernel for a product, please review the product Release Notice. If the product changes kernel libraries, please consult Technical Support before continuing.
15. The installation of the ADDEN group overwrites **/unix** and **/etc/master**. (Note that **/etc/master** is not treated as a customizable file by CTIX installations.) If you have a custom **/unix** (you changed the **dfile** or libraries and have done a kernel build), you might have to reapply those customizations after this installation.
16. *termio(7)* generation of fill characters to effect mechanical movement delays for printers and other such devices has some subtleties.

In 5.25.X CTIX and 6.X CTIX, and AT&T System V Release 3.2 UNIX the values and algorithms used are identical. The method used assigns a count value for each kind of delay. This value then does double duty. If OFILL is not set, this count is passed to "timeout" as the number of 1/60 second ticks to delay. If OFILL is set, the following algorithm is applied to calculate the number of fill characters to send.

```
if (count == 0 | count >= 32) send 0
if (count <= 3) send 1
else send 2
```

In some cases the number of fills generated disagrees with the CTIX *termio(7)* and SVID Issue 2 documentation.

In AT&T System V Release 4.0 an attempt was made to make the delays and fills correspond correctly to the documentation. Here the count value used is the actual number of fill characters to send and a different count value is set explicitly for "timeout" counts.

The following table lists the values used:

Delay type	System V Release 3.2 count	System V Release 3.2 fills	SVID fills	System V Release 4.0 fills
CR1	Note 1	2	2	2
CR2	5	2	4	4
CR3	9	2	undefined	0
NL1	5	2	2	2
TAB1	Note 2	0 or 2	2?	2
TAB2	2	1	2	2
TAB3	0	0	undefined	0
BS1	2	1	1	2
VT1	127	0	undefined	0
FF1	127	0	undefined	0

Note 1: count = MAX(column > 4 + 3, 6)

Note 2: pos = 8 - column & 0x7; if (pos < 5) count = 0
else count = pos;

Notice that System V Release 3.2 differs from SVID Issue 2 for the number of fills generated for CR2 and TAB2. The TAB1 case is unclear. According to SVID, the delay for TAB1 is column dependent and 2 fill characters will be sent for any delay, if OFILL is set. No more is stated, therefore the System V Release 3.2 interpretation is arguably correct for TAB1.

In light of the foregoing, in 6.2 CTIX the System V Release 3.2 defacto standard fills will be adhered to for the following reasons:

1. There is a potential to break existing applications by changing functionality. Our previous releases behave this way and most System V UNIX releases act this way.
2. Since the count value does double duty it would require fairly substantial code changes to effect SVID compliance without affecting "timeout" delays.

3. We will be in synch with the rest of the industry by addressing this issue in the System V Release 4.0 environment.
17. Some serial printers may lose the last lines of output. This can be corrected by placing the following line at the end of the printers interface script:

```
sleep 15
```

18. If using the second parallel printer port, /dev/plp1 the first parallel port, /dev/plp must first be initialized (opened) before any access is made to the second port. Failure to do so will result in a system panic. The first parallel port may be initialized by placing the following in /etc/rcopts/PLP:

```
lpset -4 -c132 -l66
```

19. On a system with two parallel ports (i.e., one on the CPU board and one on an IOP or 422 board), administrators might expect the printer-configuration menus of *adman(1)* to install initialization commands for both parallel ports in the file */etc/rcopts/PLP*. Instead, regardless of the *adman* options selected, only one initialization command is ever installed in */etc/rcopts/PLP*, and that command references the port on the CPU board. If any initialization commands for the second port are desired, they must be installed by hand, and such commands are subject to being overwritten by subsequent invocations of *adman(1)*.
20. The potential for lost mail exists if the *mailx(1)* program is run non-interactively, and particularly if multiple simultaneous invocations of *mailx* attempt to address the same recipient.
Note that this condition might arise inadvertently, for example, in a shell script which uses group aliases containing members in common.

7. Documentation Updates

There is a Documentation Update for this Addendum. It is included in the Addendum package. The man pages that have changed are:

1. `crash(1M)` - examine system images (11p) - bug fixes, IOP16 support, and a new command `drvtable`, which prints the loadable drivers table.
2. `hinv(1M)` - hardware inventory (2p) - IOP16 support.
3. `iopdump(1M)` - upload a Front-end I/O Processor's RAM (1p) - a new utility, so this is the first version of a man page.
4. `serstat(1M)` - display serial port error statistics (2p) - IOP16 support.
5. `gettimeofday(2)` - get/set date and time (2p) - add a warning about linking with `libsocket.a`.
6. `notify(2)` - manage notifications (5p) - changed to work in a STREAMS environment.
7. `monitor(3C)` - prepare execution profile (2p) - a clarification of the use of `monitor`.
8. `sleep(3C)` - suspend execution for interval (1p) - a warning is added about the treatment of signals by `sleep(3c)`, and use of `longjmp()` by `sleep(3C)`, which might cause premature preemption and loss of context from other nested signal handler routines.

Appendix A: Port Selection

The ports on the IOP16 board **do not** follow the same numeric sequence as the other RS232 ports on the system.

1. An IOP16 Board in slot expansion slot 1 controls tty ports 064 through 079. The following is a list of entries in the `/etc/inittab` file for those ports:

```
064:23:off:/etc/getty tty064 9600
065:23:off:/etc/getty tty065 9600
066:23:off:/etc/getty tty066 9600
067:23:off:/etc/getty tty067 9600
068:23:off:/etc/getty tty068 9600
069:23:off:/etc/getty tty069 9600
070:23:off:/etc/getty tty070 9600
071:23:off:/etc/getty tty071 9600
072:23:off:/etc/getty tty072 9600
073:23:off:/etc/getty tty073 9600
074:23:off:/etc/getty tty074 9600
075:23:off:/etc/getty tty075 9600
076:23:off:/etc/getty tty076 9600
077:23:off:/etc/getty tty077 9600
078:23:off:/etc/getty tty078 9600
079:23:off:/etc/getty tty079 9600
```

2. An IOP16 Board in slot expansion slot 2 controls tty ports 080 through 095. The following is a list of entries in the `/etc/inittab` file for those ports:

```
080:23:off:/etc/getty tty080 9600
081:23:off:/etc/getty tty081 9600
082:23:off:/etc/getty tty082 9600
083:23:off:/etc/getty tty083 9600
084:23:off:/etc/getty tty084 9600
085:23:off:/etc/getty tty085 9600
086:23:off:/etc/getty tty086 9600
087:23:off:/etc/getty tty087 9600
088:23:off:/etc/getty tty088 9600
089:23:off:/etc/getty tty089 9600
090:23:off:/etc/getty tty090 9600
091:23:off:/etc/getty tty091 9600
092:23:off:/etc/getty tty092 9600
```

093:23:off:/etc/getty tty093 9600
094:23:off:/etc/getty tty094 9600
095:23:off:/etc/getty tty095 9600

3. An IOP16 Board in slot expansion slot 3 controls tty ports 096 through 111. The following is a list of entries in the `/etc/inittab` file for those ports:

096:23:off:/etc/getty tty096 9600
097:23:off:/etc/getty tty097 9600
098:23:off:/etc/getty tty098 9600
099:23:off:/etc/getty tty099 9600
100:23:off:/etc/getty tty100 9600
101:23:off:/etc/getty tty101 9600
102:23:off:/etc/getty tty102 9600
103:23:off:/etc/getty tty103 9600
104:23:off:/etc/getty tty104 9600
105:23:off:/etc/getty tty105 9600
106:23:off:/etc/getty tty106 9600
107:23:off:/etc/getty tty107 9600
108:23:off:/etc/getty tty108 9600
109:23:off:/etc/getty tty109 9600
110:23:off:/etc/getty tty110 9600
111:23:off:/etc/getty tty111 9600

4. An IOP16 Board in slot expansion slot 4 controls tty ports 112 through 127. The following is a list of entries in the `/etc/inittab` file for those ports:

112:23:off:/etc/getty tty112 9600
113:23:off:/etc/getty tty113 9600
114:23:off:/etc/getty tty114 9600
115:23:off:/etc/getty tty115 9600
116:23:off:/etc/getty tty116 9600
117:23:off:/etc/getty tty117 9600
118:23:off:/etc/getty tty118 9600
119:23:off:/etc/getty tty119 9600
120:23:off:/etc/getty tty120 9600
121:23:off:/etc/getty tty121 9600
122:23:off:/etc/getty tty122 9600
123:23:off:/etc/getty tty123 9600
124:23:off:/etc/getty tty124 9600

125:23:off:/etc/getty tty125 9600
126:23:off:/etc/getty tty126 9600
127:23:off:/etc/getty tty127 9600

1
2
3

4

5