

1 Allgemeines

Hiermit wird das Betriebssystem SINIX V5.22 fuer den MX500 freigegeben.

SINIX V5.22 ist auf allen Modellen des MX500 ablauffaehig.

Die Freigabemitteilung gibt eine gedraengte Zusammenstellung aller wesentlichen Informationen zur vorliegenden Version 5.22 von SINIX. Mit SINIX V5.22 ist die Kompatibilitaet zum X/OPEN Portability Guide III erreicht.

Der Inhalt dieser Freigabemitteilung entspricht dem Freigabestand 5/90.

Es stehen folgende Software-Liefereinheiten zur Verfuegung:

SINIX-UP-F1	Update von V5.21 auf V5.22	(1-16 Benutzer)
SINIX-UP-F3	Update von V5.21 auf V5.22	(1-32 Benutzer)
SINIX-UP-F5	Update von V5.21 auf V5.22	(1-64 Benutzer)
SINIX-F1	SINIX-Basissystem V5.22	(1-16 Benutzer)
SINIX-F2	SINIX-Hochruestsatz fuer	17 bis 32 Benutzer
SINIX-F3	SINIX-Basissystem V5.22	(1-32 Benutzer)
SINIX-F4	SINIX-Hochruestsatz fuer	33 bis 64 Benutzer
SINIX-F5	SINIX-Basissystem V5.22	(1-64 Benutzer)
SINIX-F6	SINIX-Hochruestsatz fuer	65 bis 128 Benutzer
SINIX-F7	SINIX-Basissystem V5.22	(1-128 Benutzer)

Zusaetzlich zur gewuenschten Software-Liefereinheit muss ein Dokumentationspaket bestellt werden. Die Auflistung der verfuegbaren Liefereinheiten und der darin enthaltenen Manuale finden Sie im Kapitel 1.3.

1.1 Bestellung

SINIX Version 5.22 kann ueber Ihre zustaeendige Vertriebsgesellschaft bezogen werden.

Es muss die Softwareliefereinheit plus Dokumentationspaket bestellt werden.

Es gelten die allgemeinen Bedingungen zum Vertrag ueber die Nutzung und Betreuung von Softwareprodukten.

Die Liefereinheit ist ein Lizenzprodukt. Die Lizenznummer wird durch Installation der Diskette K.DISK (KEY-Diskette) ins System uebernomen. Dieses Produkt darf nur auf dem Geruet eingesetzt werden, fuer das es gekauft wurde.

1.2 Auslieferung

Das Betriebssystem SINIX Version 5.22 wird auf 3 Magnetbandkassetten (SINIX3, SINIX5 und SIE2.1) plus KEY-Diskette (K.DISK) ausgeliefert. Bei Update-Versionen entfaellt die KEY-Diskette, da sie von der Vorgaengerversion uebernommen werden kann.

Lieferumfang:

Kassetten SINIX3/SINIX5:

System V Environment	(X/OPEN Umgebung von SINIX V5.22)
CES Environment	(C-Entwicklungs-System V5.22)
COLLAGE	(COLLAGE-Laufzeitsystem V3.0)
COLFACE	(COLLAGE-Bediensystem)
Games	(Spiele mit englischer Oberflaeche)
SDT	(System-Development-Toolkit)

Kassette SIE2.1:

SINIX 2.1 Environment	(SINIX V2.1 Umgebung von SINIX V5.22)
-----------------------	---------------------------------------

1.3 Dokumentation

Zur Installation und zum Betrieb von SINIX V5.22 ist das Dokumentationspaket erforderlich.

Es stehen zwei Liefereinheiten zur Verfuegung:

SINIX-DOC-D V5.22	deutsche Dokumentation
SINIX-DOC-GB V5.22	englische Dokumentation

Folgende Manuale gehoeren zum Lieferumfang des Dokumentationspakets:

Bestellnummer

Betriebsanleitung MX500	U5006-J-Z95-2
SINIX Systemverwaltung V5.22	U3904-J-Z95-4
SINIX V5.22 Kommandos, Teil 1	U5453-J1-Z95-2
SINIX V5.22 Kommandos, Teil 2	U5454-J1-Z95-2
SINIX V5.22 Kommandos, Teil 3	U5455-J1-Z95-2
C-Entwicklungssystem, Teil 1	U3899-J-Z95-3
C-Entwicklungssystem, Teil 2	U3900-J-Z95-3
SINIX-SPOOL Benutzerhandbuch	U5650-J-Z95-1
COLLAGE Bediensystem Benutzerhandbuch	U5995-J-Z95-1
inkl. Kurzbeschreibung	U6007-J-Z95-1
COLLAGE Bedienen, Verwalten, Programmieren	U3004-J-Z95-4
Nachtrag zu COLLAGE Bedienen	U3004-J1-Z95-5
SINIX-Systemsicherheit	U5069-J-Z95-1
SINIX Buch1	U3201-J-Z95-1
SINIX Buch2	U3202-J-Z95-1
Nachtrag zu SINIX Buch2	U3202-J1-Z95-2

Das Kapitel Systemverwaltung im SINIX Buch2 ist nicht mehr gueltig; es wurde ersetzt durch das Kapitel Bediensystem im Systemverwalterhandbuch!

Fuer den Einsatz von Peripherie-Geraeten sind die entsprechenden Hardware-Manuale erforderlich.

2 Technische Hinweise

Auf SINIX Version 5.22 koennen unter Beachtung bestimmter Einschränkungen die auf anderen SINIX-Versionen erstellten Programme im Binaerformat portiert werden (vgl. 2.6).

Neue und erweiterte Software-Funktionen werden im Kapitel 3 dargestellt.

2.1 Hardwareausbau

MX500-75:

Minimalausbau:

- Konsol- und Diagnoseprozessor (SCED)
- Dualprozessor mit 2 x CPU NS32532 mit integrierter MMU, einschliesslich FPU und 64 KByte Cache pro Prozessor
- 1 Multibus
- 1 Speichercontroller mit 16 MB
- 5 1/4 Zoll Diskettenlaufwerk mit Steuerung
- 5 1/4 Zoll Magnetbandkassettenlaufwerk mit Steuerung, Kapazitaet 155 MB
- 1 Festplatte 5 1/4 Zoll, Kapazitaet 380 MB oder 760 MB
- 1 E/A-Prozessor oder
- 1 serieller Inhouse-Multiplexer (SIM)

Erweiterungen:

- bis zu 5 Festplatten 5 1/4 Zoll, Kapazitaet 760 MB
- bis zu 3 Dualprozessoren
- 1 Speichercontroller mit 16 MB
- bis zu 2 Speichererweiterungen mit je 16 MB
- ein zusaetzlicher Multibus
- bis zu 7 E/A-Prozessoren oder
- bis zu 7 serielle Inhouse-Multiplexer (SIM)
- bis zu 5 ladbare DFUE-Prozessoren
- 1 Magnetbandgeraet 1/2 Zoll
- 1 Magnetbandkassettenlaufwerk Video-8, Kapazitaet 2,3 GB
- 1 Ethernet-Anschluss fuer TCP/IP Protokolle
- bis zu 2 Ethernet-Prozessoren fuer ISO Protokolle
- AFP-Anschlusszusaetze fuer maximal 44 Anschuesse ueber AFP
- 1 Erweiterungsschrank mit 1 Multibus zum Einbau von
- bis zu 6 Festplatten 5 1/4 Zoll, Kapazitaet 760 MB
- bis zu 8 E/A-Prozessoren oder
- bis zu 8 seriellen Inhouse-Multiplexern (SIM)
- bis zu 4 DFUE-Prozessoren
- 1 Ethernet-Prozessor fuer ISO Protokolle
- AFP-Anschlusszusaetzen fuer max. 48 Anschuesse ueber AFP
- zusaetzlichen Peripherie-Steuerungen

Vorsicht: Nicht alle Erweiterungen sind gleichzeitig einsetzbar!

Hauptspeicherbedarf von SINIX V5.22:

Der Hauptspeicherbedarf des Betriebssystems SINIX V5.22 ist abhaengig vom Hauptspeicherausbau und laesst sich berechnen aus der Differenz der Groessen real mem minus avail mem, die beim Hochfahren des Systems ausgegeben werden.

2.2 Software-Konfiguration

Mit der Installation von SINIX Version 5.22 stehen auf Ihrem System drei Ablauf-Umgebungen zur Verfuegung:

Kommandos in der att-Ablaufumgebung:

Das att-Universum bietet Ihnen eine zu X/OPEN Portability Guide III kompatible Ablaufumgebung. Dies ist weitgehend die Programmierumgebung nach der "UNIX System V Interface Definition" (UNIX ist ein eingetragenes Warenzeichen von AT&T).

Kommandos in der sie-Ablaufumgebung:

Dies ist die SINIX V2.1-Umgebung.

Vorsicht: Die sie-Ablaufumgebung enthaelt keine Entwicklungsumgebung mehr und muss separat von der beiliegenden Magnetbandkassette SIE2.1 installiert werden!

Kommandos zur Systemadministration:

Die Systemverwalterkommandos sind systemspezifisch und unterscheiden sich z.T. von denen anderer SINIX-Versionen. Alle Systemverwalterkommandos finden Sie im Systemverwalterhandbuch beschrieben. Meldungen dieser Kommandos werden in englischer Sprache ausgegeben.

Alle weiteren Kommandos oder Systemdateien, die weder im Systemverwalterhandbuch beschrieben sind noch zu den Kommandos des sie- bzw. att-Universums gehoeren, sind als Zusatz zum SINIX V5.22 Betriebssystem zu betrachten, fuer den keine Wartung uebernommen wird.

Das C-Entwicklungssystem fuer das att-Universum ist in die Liefereinheit SINIX Version 5.22 integriert. Das C-Entwicklungssystem der SINIX V2.1 (sie-Universum) entfaellt. Durch Einsatz des Softwareprodukts DFS werden verteilte Dateisysteme unterstuetzt. DFS erfordert den Einsatz eines neuen SCED-Boards. Genaue Information hierueber erhalten Sie bei Ihrer zustaendigen Vertriebsgesellschaft.

SINIX V5.22 unterstuetzt in Verbindung mit dem Softwareprodukt COLLAGE Grafik auf Grafik- und Alpha-Bildschirmen.

Das COLLAGE-Laufzeitsystem ist Bestandteil der SINIX V5.22.

2.3 Produkt-Installation

2.3.1 Allgemeine Hinweise

SINIX V5.22 kann nur auf einer der ersten 8 Festplatten installiert werden!

Bitte ueberpruefen Sie vor der Installation den Typ der Prozessoren, mit denen Ihre MX500 ausgestattet ist. Fuer die Prozessoren NS32532 verwenden Sie dann zur Installation von SINIX V5.22 das Band SINIX5. Fuer die Prozessoren NS32032 und NS32332 verwenden Sie bitte das Band SINIX3!

Saemtliche, die Konfiguration von Bildschirmen und Druckern betreffenden Dateien duerfen nicht von aelteren SINIX-Versionen uebernommen werden, da sich die Verwaltung der konfigurierten Geraete grundlegend geaendert hat. Es muessen in jedem Fall alle Geraete neu konfiguriert werden.

Die Dateien /etc/passwd und /etc/group duerfen ebenfalls nicht von alten Sicherungsstaenden ueberschrieben werden, da sich deren Inhalt teilweise geaendert hat. Fuer den Fall, dass alte Benutzerkennungen uebernommen werden sollen, duerfen diese nur von Hand in die neue /etc/passwd kopiert werden; analog muss bei den Gruppeneintraegen vorgegangen werden. Aus Kompatibilitaetsgruenden werden auch noch die Dateien login.dat und deauth.dat im Dateiverzeichnis /usr/sie_root/usr/menus/app/develop und das Dateiverzeichnis /usr/admin/.benutzer gepflegt; diese Dateien muessen dann also von der vorherigen Installation uebernommen werden.

Die Datei /etc/rc.local darf nicht von einer aelteren SINIX-Version uebernommen werden, da sich deren Inhalt geaendert hat.

Sie sollten den vom System benutzten usr-Bereich und den von den Benutzerdaten belegten Bereich moeglichst auf verschiedenen Dateisystemen auf getrennten Platten halten. Ausserdem sollten root- und usr-Bereich auf getrennten Platten liegen.

Die Groesse des gesamten Swap-Bereichs sollte zwischen 2 und 5 mal so gross gewaehlt werden wie die Groesse des Hauptspeichers. Die guenstigste Groesse des Swap-Bereichs ist die ca. 4-fache Groesse des Hauptspeichers.

Der Swap-Bereich, der zur Aufnahme des Hauptspeicherabzugs im Falle eines Systemabsturzes benutzt wird, muss mindestens die Groesse des Hauptspeichers plus 4 MB haben. Fuer den Fall, dass dies nicht moeglich ist, kann man das bsu-Kommando dump mit entsprechenden Schaltern veranlassen, einen Hauptspeicherabzug auf mehrere Geraete zu verteilen (vgl. Systemverwalterhandbuch).

Wichtiger Hinweis zum Einrichten von Dateisystemen:
 Standardmaessig wird beim Einrichten eines Dateisystems mittels `/etc/newfs` pro 2 KB ein Inode eingerichtet, d.h., die maximale Anzahl der Inodes in einem Dateisystem richtet sich nach der Groesse des Dateisystems. Das Problem hierbei ist, dass der Systemkern in der sie-Ablaufumgebung nur maximal 65535 Inodes pro Dateisystem verwalten kann, die Anzahl der von `/etc/newfs` angelegten Inodes jedoch bei Verwendung eines sehr grossen Plattenbereichs diesen Wert uebersteigt. An dieser Stelle muss der Anwender selber dafuer sorgen, dass es nicht zu Problemen kommt; dazu ist der `/etc/newfs` mit dem Schalter `-i` aufzurufen, der angibt, fuer wieviele KB jeweils ein Inode angelegt werden soll.

Beispiel:

```
/etc/newfs /dev/xp5c liefert u.a. :
```

```
549400 sectors in 820 cylinders of 10 tracks, 67 sectors
281.3Mb in 52 cyl groups (16 c/g, 5.49Mb/g, 2048 i/g)
```

Multipliziert man die Anzahl der Zylindergruppen mit der Anzahl der Inodes pro Zylindergruppe, so erhaelt man die Anzahl der angelegten Inodes (hier: $52 * 2048 = 106496$). Da dieser Wert zu gross ist (>65535), waehlt man dann z.B. 5 KB fuer einen Inode:

```
/etc/newfs -i 5120 /dev/xp5c
```

```
549400 sectors in 820 cylinders of 10 tracks, 67 sectors
281.3Mb in 52 cyl groups (16 c/g, 5.49Mb/g, 1024 i/g)
```

Jetzt liefert die Multiplikation $52 * 1024 = 53248$; da dieser Wert kleiner ist als 65535, wird die Anzahl der maximal moeglichen Inodes fuer den Kern im sie-Universum nicht mehr ueberschritten.

Fuer dieses Beispiel wurde die c-Partition einer 300 MB Platte verwendet, d.h. die Werte fuer andere Bereiche, deren Groesse Sie selbst bestimmen koennen, werden sich von denen des Beispiels unterscheiden.

Auf einer 8 Zoll Festplatte mit einer Kapazitaet von 600 MB tritt in diesem Zusammenhang ein weiteres Problem auf. In einer Partition duerfen pro Zylindergruppe nicht mehr als 2048 Inodes eingerichtet werden und bei einer Dateisystem-Blockgroesse von 8 KB muessen mindestens 16 Zylinder (oder ein Vielfaches davon) zu einer Zylindergruppe zusammengefasst werden. Dies hat zur Folge, dass fuer einen relativ grossen Plattenbereich nur verhaeltnismaessig wenige Inodes zur Verfuegung stehen, d.h. die Partition waere bei einer Dateigroesse zwischen 6 und 7 KB optimal ausgenutzt. Bei Anwendern, die z.B. ueberwiegend mit HIT arbeiten, also sehr viele sehr kleine Dateien zu verwalten haben, kann die Plattenkapazitaet nur zur Haelfte oder weniger ausgenutzt werden.

Fuer diesen Fall wurde in SINIX V5.22 Abhilfe geschaffen, indem dem Anwender ermoeeglicht wird, vor der Systeminstallation mittels des Programms installsys festzulegen, ob er mit einem 4 KB oder einem 8 KB Dateisystem arbeiten moechte. Bei Verwendung eines 4 KB Dateisystems kann naemlich mit 8 Zylindern pro Zylindergruppe gearbeitet werden, wodurch sich die Anzahl der Inodes ungefaehr verdoppelt.

Einrichten des root-Dateisystems

Auf einer 8 Zoll Festplatte mit einer Kapazitaet von 600 MB sollte grundsaeztlich fuer das root-Dateisystem die Block-groesse 4 KB gewaehlt werden, da sonst die Anzahl der verfueg-baren Inodes im Dateisystem nicht gross genug ist.

Einrichten des usr-Dateisystems

Falls die Anzahl der Inodes auf der Partition, auf der Sie den usr-Bereich einrichten wollen, groesser als 65535 wird, sollten Sie bei der Systeminstallation vor dem Aufruf der Prozedur restore.more die hiervon implizit aufgerufene Prozedur restore.more.sh dahingehend abaendern, dass Sie in den Aufruf des Kommandos /etc/newfs den Schalter -i mit dem entsprechenden Wert einfuegen. Dies funktioniert zum Zeitpunkt der Installation nur mit dem Editor ed, da keiner der anderen Editoren zu diesem Zeitpunkt verfuegbar ist. Folgende Kommandofolge bewirkt die notwendige Aenderung:

```
# cd /etc
# ed restore.more.sh
.
.
/newfs/s/newfs/newfs -i xxxx/p
w
q
# cd /
```

Hierbei ist xxxx durch einen geeigneten Wert (s.o.) zu ersetzen.

Anschliessend wird die Prozedur restore.more gestartet, wie in der Installationsanleitung beschrieben.

Vor der eigentlichen Installation des Betriebssystems SINIX V5.22 haben Sie mittels des Programms installsys die Moeglichkeit, die Aufteilung aller vorhandenen Platten frei festzulegen. Das Programm installsys erwartet die Platten-namen in "Monitornotation" (z.B. xp(0,0) fuer die erste xp-Platte). Die Formel fuer die Berechnung der einzelnen Platten lautet fuer Platten im Multibus:

$$512 * \text{Multibusadapternr.} + 8 * \text{Controllernr.} + \text{Laufwerksnr.}$$

und fuer Platten am SCSI-Bus (sd-Platten):

```
8 * target adapter nr. + Laufwerksnr. .
```

Fuer SINIX V5.22 ist das Standalone-Dienstprogramm installsys dahingehend erweitert worden, dass jetzt bei der Festlegung der Plattenaufteilung fuer jede Partition festgelegt werden kann, ob sie ein 4 KB Dateisystem oder ein 8 KB Dateisystem aufnehmen soll. Dies ist vor allem fuer die root-Partition einer 8 Zoll Platte mit 600 MB notwendig (s.o.).

Vor der Installation des Betriebssystems ueberpruefen Sie bitte noch, ob alle gewuenschten Zusatz-Softwareprodukte die notwendigen Versionsstaende haben (vgl. Kapitel 2.7.2).

2.3.2 Installation von SINIX V5.22 auf is-Platte

Das Promptzeichen des Monitors ist der *.

```
*b 0 ts(32,4)      Laden des Standalone-Dienstprogramms
                    installsys
```

```
Mem: x.x M
```

```
chk .....
```

```
mem .....
```

```
test
```

```
.
```

```
.
```

```
.          Eroeffnungsbild des Programms installsys
.
```

```
device? is(0,2)
```

Mit Hilfe des Programms installsys muss vor der eigentlichen Installation die Plattenaufteilung festgelegt werden. Dabei muss fuer den root-Bereich mindestens der Standardwert von 12 MB und fuer den Swap-Bereich der Standardwert von 36 MB verwendet werden. Die Mindestgrosse des Swap-Bereichs sollte im allgemeinen die Grosse des Hauptspeichers plus 4 MB betragen.

Die weitere Aufteilung der Platte ist frei waehlbar, wobei jedoch darauf geachtet werden muss, dass ein Plattenbereich (e, f, g oder h) gross genug fuer die Installation des /usr-Bereichs ist. Nachdem die Aenderungen fuer die erste Platte in das Label geschrieben wurden, koennten Sie, wenn Sie auf die Frage "Another disk?" mit "y" antwor-

ten, die Aufteilung weiterer Platten veraendern. Antworten Sie hier mit "n", so wird mit der Installation des Miniroot-Systems in der Partition b der ersten Platte begonnen.

Input file? ts(32,13) Das Miniroot-Filesystem ist die Datei Nr. 13 auf dem Installationsband.

Output file? is(0,1) is(0,1) bezeichnet die Swap-Partition der ersten Platte.

Last chance.... RETURN

Jetzt kopiert installsys die Daten und berechnet eine Pruefsumme.

>>> EOF Pass 1, count "xxxxx", checksum "xxxx",
doing verify
Pass 2, count "xxxxx", checksum "xxxx"

Done
exit 0

Das Programm installsys ist beendet.

* b 3 ts(32,0) Laden des Standalone-Dienstprogramms boot. Dies ist Datei Nr. 0 auf dem Installationsband

Boot

:is(0,1)gendynix is0* Hiermit wird der generische Kern des Betriebssystems von der Platte in den Hauptspeicher geladen. Der * nach is0 sagt gendynix, dass ein Teil der Partition b der Platte fuer das Miniroot- Dateisystem verwendet werden soll und ein Teil als swap-Partition.

Anschliessend werden eine Reihe von Meldungen ueber die automatische Konfiguration des Kerns auf der Konsole ausgegeben. Zum Schluss gibt SINIX V5.22 den Prompt fuer den Superuser aus (#).

Sie arbeiten jetzt mit einer abgemagerten Version des Betriebssystems SINIX V5.22. Als naechstes muessen Sie mit Hilfe des Programms restore.root ein Dateisystem in der Partition a der 1. Platte erzeugen und das Standard-root-Dateisystem dorthin kopieren.

```
# restore.root
.
.
.
Build the root file system
disk name for root? (e.g. sd0 or xp0 or is0) is0
```

restore.root gibt folgende Meldung an der Konsole aus:

Press the AUTOBOOT button so an automatic reboot can happen.

Die Prozedur restore.root richtet in der Partition a der oben angegebenen Platte ein neues Dateisystem ein und kopiert das root-Dateisystem vom Band dorthin. Anschliessend wird die abgemagerte Version von SINIX V5.22 beendet und wenn die Taste AUTOBOOT gedrueckt ist wird automatisch die Standardversion von SINIX V5.22 von der Partition a der Platte geladen.

Mit Hilfe der Prozedur restore.more koennen Sie jetzt das usr-Dateisystem erstellen und die entsprechenden Programme, Kommandos, etc. vom Band kopieren. Bevor Sie dies tun, sollten Sie mit /etc/newfs ein Dateisystem auf der Plattenpartition einrichten, die Sie fuer den usr-Bereich verwenden wollen, und dabei ueberpruefen, ob die Anzahl der Inodes unter der Grenze von 65535 bleibt (vgl. Kapitel "Wichtiger Hinweis zum Einrichten von Dateisystemen").

```
# restore.more
restore /usr ?(n/y) y
disk name for /usr? (e.g. sd0 or xp0 or is0) is1
which partition for /usr? (e, f, g or h) h
```

Die Prozedur restore.more fragt, auf welcher Platte das usr-Dateisystem erzeugt werden soll. Hier sollte eine andere als die erste Platte angegeben werden.

Im Anschluss an diese Fragen erhalten Sie die Moeglichkeit festzulegen, welche Dateien und Dateiverzeichnisse Sie auf die Platte kopieren wollen.

```
restore /usr/games ? (n/y) y
restore CES Environment ? y
```

restore.more erzeugt in der Partition h der Platte ein Dateisystem und kopiert das usr-Dateisystem und die Ablaufumgebung att dorthin. Zusaetzlich koennen Sie entscheiden, ob Sie auch die C-Entwicklungsumgebung installieren moechten.

Nach der Installation von SINIX V5.22 muessen Sie das System herunterfahren. Beim erneuten Hochfahren wird dann die mitgelieferte KEY-Diskette (K.DISK) mit der Meldung "Please install your SINIX authorization file" angefordert, mit der Ihr SINIX-Betriebssystem einsatzbereit gemacht wird.

Auf Ihrem MX500 ist jetzt das Betriebssystem SINIX V5.22 komplett installiert.

Das Eintragen von Benutzerkennungen und die Konfigurierung von Terminals und Druckern erfolgt wie gewohnt unter der Kennung admin, jedoch ueber das neue Bediensystem. Eine Bedienungsanleitung hierzu finden Sie im Systemverwalterhandbuch. Zusaetzlich zu mountende Dateisysteme tragen Sie bitte von Hand in die Datei /etc/fstab ein.

2.3.3 Installation von SINIX V5.22 auf xp-Platte

Das Promptzeichen des Monitors ist der *.

```
*b 0 ts(32,4)      Laden des Standalone-Dienstpro-
                   gramms installsys
```

```
Mem: x.x M
```

```
chk .....
```

```
mem .....
```

```
test
```

```
.
```

```
.
```

```
.          Eroeffnungsbild des Programms in-
.          stallsys
```

```
device? xp(0,2)
```

Mit Hilfe des Programms installsys muss vor der eigentlichen Installation die Plattenaufteilung festgelegt werden. Dabei muss fuer den rot-Bereich mindestens der Standardwert von 12 MB und fuer den Swap-Bereich der Standardwert von 36 MB verwendet werden. Die Mindestgrosse des Swap-Bereichs sollte im allgemeinen die Grosse des Hauptspeichers plus 4 MB betragen.

Die weitere Aufteilung der Platte ist frei waehlbar, wobei jedoch darauf geachtet werden muss, dass ein Plattenbereich (e, f, g oder h) gross genug fuer die Installation des /usr-Bereichs ist. Nachdem die Aenderungen fuer die erste Platte in das Label geschrieben wurden, koennten Sie, wenn Sie auf die Fra-

ge "Another disk?" mit "y" antworten, die Aufteilung weiterer Platten veraendern. Antworten Sie hier mit "n", so wird mit der Installation des Miniroot-Systems in der Partition b der ersten Platte begonnen.

Input file? ts(32,13) Das Miniroot-Filesystem ist die Datei Nr. 13 auf dem Installationsband.

Output file? xp(0,1) xp(0,1) bezeichnet die Swap-Partition der ersten Platte.

Last chance.... RETURN

Jetzt kopiert installsys die Daten und berechnet eine Pruefsumme.

>>> EOF Pass 1, count "xxxxx", checksum "xxxx",
doing verify
Pass 2, count "xxxxx", checksum "xxxx"

Done
Exit 0

Das Programm installsys ist beendet.

*b 3 ts(32,0) Laden des Standalone-Dienstprogramms boot. Dies ist Datei Nr. 0 auf dem Installationsband

Boot

:xp(0,1)gendynix xp0* Hiermit wird der generische Kern des Betriebssystems von der Platte in den Hauptspeicher geladen. Der * nach xp0 sagt gendynix, dass ein Teil der Partition b der Platte fuer das Miniroot- Dateisystem verwendet werden soll und ein Teil als swap-Partition.

Anschliessend werden eine Reihe von Meldungen ueber die automatische Konfiguration des Kerns auf der Konsole ausgegeben. Zum Schluss gibt SINIX V5.22 den Prompt fuer den Superuser aus (#).

Sie arbeiten jetzt mit einer abgemagerten Version des Betriebssystems SINIX V5.22. Als naechstes muessen Sie mit Hilfe des Programms restore.root ein Dateisystem in der Partition a der 1. Platte erzeugen und das Standard-root-Dateisystem dorthin kopieren.

```
# restore.root
.
.
.
Build the root file system
disk name for root? (e.g. sd0 or xp0 or is0) xp0
```

restore.root gibt folgende Meldung an der Konsole aus:

Press the AUTOBOOT button so an automatic reboot can happen.

Die Prozedur restore.root richtet in der Partition a der oben angegebenen Platte ein neues Dateisystem ein und kopiert das root-Dateisystem vom Band dorthin. Anschliessend wird die abgemagerte Version von SINIX V5.22 beendet und wenn die Taste AUTOBOOT gedrueckt ist wird automatisch die Standardversion von SINIX V5.22 von der Partition a der Platte geladen.

Mit Hilfe der Prozedur restore.more koennen Sie jetzt das usr-Dateisystem erstellen und die entsprechenden Programme, Kommandos, etc. vom Band kopieren. Bevor Sie dies tun, sollten Sie mit /etc/newfs ein Dateisystem auf der Plattenpartition einrichten, die Sie fuer den usr-Bereich verwenden wollen, und dabei ueberpruefen, ob die Anzahl der Inodes unter der Grenze von 65535 bleibt (vgl. Kapitel "Wichtiger Hinweis zum Einrichten von Dateisystemen").

```
# restore.more
restore /usr ?(n/y) y
disk name for /usr? (e.g. sd0 or xp0 or is0) xp1
which partition for /usr? (e, f, g or h) h
```

Die Prozedur restore.more fragt, auf welcher Platte das usr-Dateisystem erzeugt werden soll. Hier sollte eine andere als die erste Platte angegeben werden.

Im Anschluss an diese Fragen erhalten Sie die Moeglichkeit festzulegen, welche Dateien und Dateiverzeichnisse Sie auf die Platte kopieren wollen.

```
restore /usr/games ? (n/y) y
restore CES Environment ? y
```

restore.more erzeugt in der Partition h der Platte ein Dateisystem und kopiert das usr-Dateisystem und die Ablaufumgebung att dorthin. Zusaetzlich koennen Sie entscheiden, ob Sie auch die C-Entwicklungsumgebung und Spiele mit englischer Benutzeroberflaeche installieren moechten.

Nach der Installation von SINIX V5.22 muessen Sie das System herunterfahren. Beim erneuten Hochfahren wird dann die mitgelieferte KEY-Diskette (K.DISK) mit der Meldung "Please install your SINIX authorization file" angefordert, mit der Ihr SINIX-Betriebssystem einsatzbereit gemacht wird.

Auf Ihrem MX500 ist jetzt das Betriebssystem SINIX V5.22 komplett installiert.

Das Eintragen von Benutzerkennungen und die Konfigurierung von Terminals und Druckern erfolgt wie gewohnt unter der Kennung admin, jedoch ueber das neue Bediensystem. Eine Bedienungsanleitung hierzu finden Sie im Systemverwalterhandbuch. Zusaetzlich zu mountende Dateisysteme tragen Sie bitte von Hand in die Datei /etc/fstab ein.

2.3.4 Installation von SINIX V5.22 auf sd-Platte (nur MX500-20 und MX500-70)

Das Promptzeichen des Monitors ist der *.

```
*b 0 ts(32,4)      Laden des Standalone-Dienstpro-
                   gramms installsys
```

```
Mem: x.x M
chk .....
mem .....
test
```

```
.
.
.      Eroeffnungsbild des Programms in-
.      stallsys
device? sd(48,2)
```

Mit Hilfe des Programms installsys muss vor der eigentlichen Installation die Plattenaufteilung festgelegt werden. Dabei muss fuer den rot-Bereich mindestens der Standardwert von 12 MB. Die Groesse des Swap-Bereichs sollte im allgemeinen die Groesse des Hauptspeichers plus 4 MB betragen.

Die weitere Aufteilung der Platte ist frei waehlbar, wobei jedoch darauf geachtet werden muss, dass ein Plattenbereich (e, f, g oder h) gross genug fuer die Installation des /usr-Bereichs ist. Nachdem die Aenderungen fuer die erste Platte in das Label geschrieben wurden, koennten Sie, wenn Sie auf die Frage "Another disk?" mit "y" antwor-

ten, die Aufteilung weiterer Platten veraendern. Antworten Sie hier mit "n", so wird mit der Installation des Miniroot-Systems in der Partition b der ersten Platte begonnen.

Input file? ts(32,13) Das Miniroot-Filesystem ist die Datei Nr. 13 auf dem Installationsband.

Output file? sd(48,1) sd(48,1) bezeichnet die Swap-Partition der ersten Platte.

Last chance... RETURN

Jetzt kopiert installsys die Daten und berechnet eine Pruefsumme.

>>> EOF Pass 1, count "xxxxx", checksum "xxxx",
doing verify
Pass 2, count "xxxxx", checksum "xxxx"

Done
exit 0

Das Programm installsys ist beendet.

* b 3 ts(32,0) Laden des Standalone-Dienstprogramms boot. Dies ist Datei Nr. 0 auf dem Installationsband

Boot

:sd(48,1)gendynix sd0* Hiermit wird der generische Kern des Betriebssystems von der Platte in den Hauptspeicher geladen. Der * nach sd0 sagt gendynix, dass ein Teil der Partition b der Platte fuer das Miniroot- Dateisystem verwendet werden soll und ein Teil als swap-Partition.

Anschliessend werden eine Reihe von Meldungen ueber die automatische Konfiguration des Kerns auf der Konsole ausgegeben. Zum Schluss gibt SINIX V5.22 den Prompt fuer den Superuser aus (#).

Sie arbeiten jetzt mit einer abgemagerten Version des Betriebssystems SINIX V5.22. Als naechstes muessen Sie mit Hilfe des Programms restore.root ein Dateisystem in der Partition a der 1. Platte erzeugen und das Standard-root-Dateisystem dorthin kopieren.


```
# restore.root
```

```
.
.
.
```

```
Build the root file system
```

```
disk name for root? (e.g. sd0 or xp0 or is0) sd0
```

restore.root gibt folgende Meldung an der Konsole aus:

Press the AUTOBOOT button so an automatic reboot can happen.

Die Prozedur restore.root richtet in der Partition a der oben angegebenen Platte ein neues Dateisystem ein und kopiert das root-Dateisystem vom Band dorthin. Anschliessend wird die abgemagerte Version von SINIX V5.22 beendet und wenn die Taste AUTOBOOT gedrueckt ist wird automatisch die Standardversion von SINIX V5.22 von der Partition a der Platte geladen.

Mit Hilfe der Prozedur restore.more koennen Sie jetzt das usr-Dateisystem erstellen und die entsprechenden Programme, Kommandos, etc. vom Band kopieren.

Bevor Sie dies tun, sollten Sie mit /etc/newfs ein Dateisystem auf der Plattenpartition einrichten, die Sie fuer den usr-Bereich verwenden wollen, und dabei ueberpruefen, ob die Anzahl der Inodes unter der Grenze von 65535 bleibt (vgl. Kapitel "Wichtiger Hinweis zum Einrichten von Dateisystemen").

```
# restore.more
```

```
restore /usr ?(n/y) y
```

```
disk name for /usr? (e.g. sd0 or xp0 or is0) sd1
```

```
which partition for /usr? (e, f, g or h) h
```

Die Prozedur restore.more fragt, auf welcher Platte das usr-Dateisystem erzeugt werden soll. Hier sollte eine andere als die erste Platte angegeben werden.

Im Anschluss an diese Fragen erhalten Sie die Moeglichkeit festzulegen, welche Dateien und Dateiverzeichnisse Sie auf die Platte kopieren wollen.

```
restore /usr/games ? (n/y) y
```

```
restore CES Environment ? y
```

restore.more erzeugt in der Partition h der Platte ein Dateisystem und kopiert das usr-Dateisystem und die Ablaufumgebung att dorthin. Zusaetzlich koennen Sie entscheiden, ob Sie auch die C-Entwicklungsumgebung installieren moechten.

Nach der Installation von SINIX V5.22 muessen Sie das System herunterfahren. Beim erneuten Hochfahren wird dann die mitgelieferte KEY-Diskette (K.DISK) mit der Meldung "Please install your SINIX authorization file" angefordert, mit der Ihr SINIX-Betriebssystem einsatzbereit gemacht wird.

Auf Ihrem MX500 ist jetzt das Betriebssystem SINIX V5.22 komplett installiert.

Das Eintragen von Benutzerkennungen und die Konfigurierung von Terminals und Druckern erfolgt wie gewohnt unter der Kennung admin, jedoch ueber das neue Bediensystem. Eine Bedienungsanleitung hierzu finden Sie im Systemverwalterhandbuch. Zusaetzlich zu mountende Dateisysteme tragen Sie bitte von Hand in die Datei /etc/fstab ein.

2.3.5 Installation der sie-Ablaufumgebung

Die sie-Ablaufumgebung ist in SINIX V5.22 nicht mehr im Betriebssystem enthalten. Sie wird auf einer separaten Kassette ausgeliefert und kann ueber /etc/superinstall installiert werden. Da die meisten der zusaetzlichen Softwareprodukte diese Ablaufumgebung nach wie vor voraussetzen, sollte diese Kassette sofort nach der Systeminstallation eingespielt werden.

2.3.6 Installation des System Development Toolkit (SDT)

Um das auf dem Installationsband enthaltene System Development Toolkit (SDT) installieren zu koennen, muss vor Aufruf der Prozedur restore.more die Umgebungsvariable SDT auf y gesetzt werden. Gehen Sie wie folgt vor:

```
# SDT=y
# export SDT
# restore.more
.
.
.
restoring SDT Environment
.
.
.
```

Sie koennen das SDT sowohl bei der Erstinstallation von SINIX V5.22 als auch nachtraeglich separat installieren.

Falls die Umgebungsvariable SDT nicht fuer die gesamte Sitzung erhalten bleiben soll, koennen sie die Prozedur restore.more auch folgendermassen aufrufen:

```
# SDT=y restore.more
```

2.3.7 Installation der internationalisierten Datenbasen

Seit SINIX V5.22 sind die internationalisierten Datenbasen auf dem Installationsband vorhanden und koennen analog zum SDT durch das Setzen der Umgebungsvariablen INTL (INTL=y; export INTL) vor dem Aufruf der Prozedur restore.more installiert werden.

2.3.8 Konfigurierte Swap-Bereiche nach der Installation

Nach der Systeminstallation ist in SINIX V5.22 nur die b-Partition der Systemplatte als Swap-Bereich im Kern eingetragen. Sollen weitere Plattenbereiche als Swap-Bereiche verwendet werden koennen, so muessen sie mit dem Dienstprogramm /etc/rootswap im Kern konfiguriert werden. Anschliessend an diese Konfiguration muss der Rechner herunter und anschliessend wieder hochgefahren werden. Eine Beschreibung des Dienstprogramms rootswap finden Sie im Systemverwalterhandbuch.

2.3.9 Installation von SINIX auf einer anderen als der ersten Platte

Bei der Installation von SINIX auf einer anderen als der ersten Platte ist darauf zu achten, dass immer die richtige Plattenbezeichnung angegeben wird. Die Monitor-Notation entnehmen Sie bitte dem Systemverwalter-Handbuch, die SINIX-Notation zaehlt alle Platten desselben Typs beginnend bei 0 hoch, wobei in SINIX V5.22 die Zuordnung der Platten zu den Controllern nicht mehr dynamisch ist (d.h., die erste Platte am zweiten Controller heisst immer is2, auch wenn am ersten Controller nur eine Platte angeschlossen ist!). Bei Platten-Mischkonfiguration (Verwendung von 80 MB und 300 MB-Platten bei einem MX500-20) ist zu beachten, dass alle sd-Platten beginnend bei 0 hochgezaehlt werden (sd0, sd1, usw.) und alle is-Platten ebenfalls bei 0 beginnend (is0, is1, usw.). Besondere Vorsicht ist beim Laden des Miniroot-Systems geboten, da hier beide Notationen gemischt werden. Fuer die fuenfte 5 1/4 Zoll Platte im Multibus (erste Platte am dritten Controller) muesste man folgendes eingeben:

```

.
.
.
b3ts(32,0) Laden des bsu-Boot-Kommandos
.
.
.
:is(528,1)gendynix is4*    Laden des Miniroot-Systems
                          aus dem Swap-Bereich der
                          fuefften Platte

```

Die Angabe is4* sagt dem gendynix, dass die Partition b der fuefften Platte (is4) sowohl fuer das Miniroot-System als auch als Swap-Bereich verwendet werden soll. Da diese Angabe ein Argument fuer gendynix ist, muss hier die SINIX-Notation verwendet werden.

2.3.10 Nachtraegliche Software-Installation von der Systemkassette

Wird Software nachtraeglich von Kassette eingelesen, so sollten die Archive, die bereits im usr-Bereich vorhanden sind, nochmals eingelesen werden. Andernfalls fehlen u.U. Links zwischen den Universen.

2.3.11 Massnahmen vor und nach einem Plattencrash

Es empfiehlt sich, nach der Installation des Systems und aller gewuenschten Software-Produkte, nach der Konfigurierung der gesamten Peripherie und nach der Eintragung aller Benutzerkennungen, jeweils eine komplette Dump-Sicherung der root- und usr-Partition anzulegen. Mit diesen Sicherungen kann dann im Falle eines Plattencrashes die Anlage schnell und ohne Probleme wieder komplett installiert werden.

Dazu beginnen Sie bitte wie bei einer normalen Systeminstallation mit dem Kopieren des Miniroot-Systems in den Swap-Bereich der Systemplatte. Anschliessend laden Sie dieses Minisystem wie in der Installationsanleitung beschrieben. Anstelle von restore.root machen Sie dann jedoch folgendes:

```

Anlegen der Datei /etc/mtab: >/etc/mtab
Einrichten des Dateiverzeichnisses /a: mkdir /a
Erzeugen eines Dateisystems fuer den root-Bereich:
/etc/newfs /dev/xp0a
"Mounten" des root-Bereichs: /etc/mount /dev/xp0a /a
Wechseln in den root-Bereich: cd /a
Einlesen der Sicherung des root-Bereichs:
/etc/restore rvf /dev/rts0

```

Anschliessend haengen Sie den root-Bereich wieder ab, fahren das System mit /etc/halt herunter und sofort wieder hoch in den Einbenutzerbetrieb. Dann verfahren Sie fuer die Ruecksicherung des usr-Bereichs entsprechend.

2.3.12 Wiederholte Installation von SINIX V5.22

Beabsichtigen Sie eine komplette Neuinstallation des Betriebssystems, so sollten Sie zuvor neben den Benutzerdaten die unten genannten wichtigen System-Dateien und Dateiverzeichnisse sichern; dies erspart Ihnen nach der Installation einige Arbeiten.

```
/etc/passwd      /usr/menus/app/develop/login.dat
/etc/group       /usr/menus/app/develop/deauth.dat
/etc/fstab       /usr/menus/app/control/auth.file
/etc/securetty   /usr/att/etc/profile
/etc/.profile    /usr/admin/.benutzer
/dev/MAKEDEV.local /usr/local
```

Vorsicht bei Versionswechsel: Bei einem Versionswechsel des Betriebssystems sollten diese Dateien nur nach genauer Pruefung uebernommen werden, da sich in diesen Faellen das interne Format der Systemdateien aendern kann.

Weitere wichtige Dateien anderer installierter Software-Produkte sollten Sie ebenfalls sichern (z.B. Nameserver fuer CMX, Netzdatenbank fuer LAN1).

Beachten Sie, dass bei einer Zweitinstallation root- und usr-Partition immer gemeinsam installiert werden muessen. Eine Installation der root-Partition alleine ist nicht moeglich, da hierbei noch nicht die fuer den spaeteren Betrieb notwendigen symbolischen Links eingerichtet werden. Sichern Sie deshalb unbedingt alle Benutzerdaten der usr-Partition, falls Sie neu installieren.

2.3.13 Installation von zusaetzlichen Software-Produkten

Die Installation zusaetzlicher Software-Produkte unter Alpha-COLLAGE sollte ausschliesslich im Full-Screen-Modus durchgefuehrt werden, da die Installationsprozeduren oft die 25. Bildschirmzeile benutzen, die nur im Full-Screen-Modus sichtbar ist.

2.4 Produkt-Einsatz

SINIX Version 5.22 ist auf allen Modellen der Rechnerfamilie MX500 ablauffaehig.

Die Anzahl der simultan betreibbaren Bildschirme haengt sehr stark vom Hauptspeicherbedarf, der I/O-Last und dem Programmaufbau ab und muss daher auf die Gegebenheiten des einzelnen Anwenders abgestimmt werden.

Unter Umstaenden koennen unter Alpha-COLLAGE nur noch die Haelfte der sonst mit der entsprechenden Anwendung betreibbaren Bildschirmarbeitsplaetze performant eingesetzt werden.

Ohne Einsatz von Alpha-COLLAGE berechnet sich die Anzahl der Grafik- und Alpha-Bildschirme wie folgt:

$$\text{Anz. Alpha-BS} + 1,5 * \text{Grafik-BS} \leq 256.$$

Anzahl der maximal mit HIT betreibbaren Terminals:

Ohne Alpha-COLLAGE: 90

mit Alpha-COLLAGE: 40

Arbeitsspeicherbedarf beim Einsatz von HIT unter Alpha-COLLAGE:

einmaliger Anteil: 5210 Bytes

Anteil pro Terminal: 810 Bytes

Die aktuell installierte SINIX-Version inklusive Nachtragsbezeichnung koennen Sie ueber den Menuepunkt "Installierte Softwareprodukte" bzw. direkt in der Datei /usr/admin/.products/SINIX ermitteln.

Kurse zu den Versionen SINIX V5.x werden an der "Schule fuer Kommunikation und Datentechnik" angeboten.

Durch die Autokonfigurationsfaehigkeit des Kerns koennen zusaetzliche Baugruppen (LAN-, DUE-, Hauptspeicher, Prozessoren) ohne Kernelaenderung und Neuinstallation eingebracht werden.

Viele Programme des att-Universums sind jetzt 8-Bit-clean.

Zur Installation von CMX ist das CES-Environment Voraussetzung.

Vor der Installation von LAN/REMOS muss die sie-Ablaufumgebung installiert sein. Nach der Installation von LAN/REMOS kann die Konfiguration und Netzeinbindung erst dann durchgefuehrt werden, wenn das COLLAGE-Bediensystem beendet und wieder neu gestartet wurde.

Ein 8-Bit-Terminal kann erst ab dem SCED-Firmwarestand 43A als Console eingesetzt werden; Fremdterminals sind generell nicht als Konsolterminals zugelassen.

Ein Grafik-Terminal darf nicht als Console verwendet werden.

Bei der Installation zusätzlicher Softwareprodukte darf die Umgebungsvariable PATH nur folgendermassen veraendert werden:

```
PATH=$PATH:<neuer Pfad>
export PATH
```

d.h., die Umgebungsvariable darf nur ergaenzt, nicht aber komplett ueberschrieben werden!

C-Programme, die unter einer Pilotversion von SINIX V5.22 uebersetzt wurden, muessen nach Einsatz der offiziellen Version erneut uebersetzt werden.

Inhaltsverzeichnis des Installationsbandes

Auf dem Installationsband befindet sich ein Inhaltsverzeichnis desselben in abdruckbarer Form (Datei Nr. 2). Sie koennen sich dieses Inhaltsverzeichnis entweder mit dem bsu-Kommando cat direkt ansehen, oder aber es mit dem SINIX-Kommando cp auf die Platte kopieren und anschliessend ausdrucken lassen. Zusaetzlich finden Sie im Anhang dieser Freigabemitteilung einen Ausdruck dieses Inhaltsverzeichnisses.

Datei /etc/disktab - Programm /etc/showblk0

Die Datei /etc/disktab ist nur noch zu Dokumentationszwecken im Betriebssystem vorhanden. Die Plattendaten, die von diversen Dienstprogrammen (z. B. /etc/newfs) benoetigt werden, werden in jedem Fall direkt aus dem Block 0 der jeweiligen Festplatte gelesen. Deshalb ist auch die Angabe des symbolischen Geraetenamens beim Kommando /etc/newfs ueberfluessig. Um die im Block 0 jeder Platte hinterlegten Daten zu lesen, existiert das Kommando /etc/showblk0, das als einzigen Aufrufparameter den Namen der betreffenden Platte (z. B. xp1) erwartet.

Kopieren von ganzen Plattenbereichen

Das physikalische Kopieren von kompletten Plattenbereichen mit dem Kommando cp ist generell moeglich. Jedoch muss hierbei beachtet werden, dass die Bereiche a und c nicht physikalisch kopiert werden duerfen, da diese Bereiche den Boot-Block und die Plattenaufteilung bzw. die Fehlstellenliste der jeweiligen Festplatte enthalten.

Plattenzuordnung zu den Controllern

Im Gegensatz zu frueheren SINIX-Versionen, in denen die Festplatten im Multibus den vorhandenen Controllern dynamisch zugeordnet wurden, besteht in SINIX V5.22 eine feste Zuordnung.

Festplatten im SCSI-Bus werden ihren Controllern hingegen dynamisch zugeordnet.

Wechsel in die Benutzerkennung admin mittels su

Soll mittels des Kommandos su in die Benutzerkennung admin gewechselt werden, so sollte der Aufruf folgendermassen aussehen:

```
su - admin
```

Hierbei wird die komplette Umgebung der Benutzerkennung admin substituiert, d.h. alle Umgebungsvariablen, etc. sind korrekt gesetzt. Die einzige Ausnahme hiervon bildet das Universum - es wird nicht auf att gesetzt.

Wechsel der Benutzerkennung mit /bin/login

Das Kommando login darf aus einer Shell heraus nur mit "exec /bin/login <benutzerkennung> aufgerufen werden, da sonst nach dem Ausloggen mittels <END>-Taste falsche Ausgaben vom Kommando who geliefert werden.

In einem COLLAGE-Fenster darf das Kommando login generell nicht aufgerufen werden.

"Verstuemmelte" Systemmeldungen auf der Console

Ist auf der Console Alpha-COLLAGE geladen, so werden Systemmeldungen immer im Grafik-Zeichensatz, also verstuemmelt" dargestellt und zerstieren ausserdem den Bildschirminhalt. Ein Bildschirm-Refresh ist danach ueber das Menu Desktop moeglich, die Systemmeldung muessen Sie sich in der Protokoll-Datei /usr/adm/messages ansehen.

Voreingestellte Systemgrenzen

```

Max. Prozessanzahl des Systems:          nproc      532
Max. Prozessanzahl pro Benutzer:         maxuprc    100
      (Ausnahmen sind "root" u. "admin")
Max. Anz. aktiver Inodes im System:       ninode     644
Max. Anz. offene Dateien im System:       nfile     1051
Max. Anz. offene Dateien pro Prozess:     64
Max. Anz. der "record locks":             lck_max    500
Max. Anz. der fuer den Buffer-Cache
      reservierten HSP-Seiten:             bufpages   *)
Maximale Anzahl Named Pipes:              fifopct    **)
Maximale Anzahl Benutzerkennungen:        300

```

*) Die Anzahl der fuer den Buffer-Cache reservierten Hauptspeicherseiten ist abhaengig von der Groesse des zur Verfuegung stehenden Hauptspeichers. Eine Hauptspeicherseite hat beim Einsatz der Prozessoren NS32032 und NS32332 die Groesse 2 KB und beim Einsatz der Prozessoren NS32532 die Groesse 4 KB. Standardmaessig belegt der Buffer Cache ungefaehr 10 % des zur Verfuegung stehenden Hauptspeichers.

**) Die Anzahl der Named Pipes ist variabel. Sie richtet sich nach der Groesse des Buffer Cache. Standardmaessig werden 66 % des Buffer Cache fuer Named Pipes verwendet. Die Variable fifopct gibt an, wieviel Prozent des Buffer-Cache fuer Named Pipes verwendet werden. Dieser Wert sollte zwischen 10 und 90 % liegen.

Veraendern der Systemgrenzen mit adb

Mit Hilfe des adb koennen die folgenden Systemgrenzen an die Beduerfnisse der Kundenanlage angepasst werden:

```

ninode
nfile
maxuprc
lck_max
bufpages
fifopct

```

```

Beispiel: # adb -w /dynix
           ninode?*w <neuer Wert>
           $q

```

Der Wert von nproc darf auf keinen Fall mit Hilfe des adb veraendert werden!

Veraendern bestimmter Systemgrenzen mit /etc/maxusers

Mit Hilfe des Dienstprogramms /etc/maxusers kann der Anwender die Variable MAXUSER und damit die Werte fuer nproc, ninode und nfile hochsetzen, da diese alle von der Variablen MAXUSER abhaengen.

Zum Aendern der Werte gehen Sie wie folgt vor:

- Abfragen des aktuellen Wertes fuer MAXUSER
/etc/maxusers
- Aendern des Wertes fuer MAXUSER
/etc/maxusers <neuer Wert>

Das Programm veraendert im Betriebssystemkern den Wert der Variablen MAXUSER und alle davon abhaengigen Parameter; deshalb muss anschliessend das System herunter und wieder hochgefahren werden, um den geaenderten Kern zu aktivieren.

Hinweis: Mit dem Programm /etc/maxusers kann nicht die Anzahl der moeglichen Logins hochgesetzt werden!

Systemauslastung

Das System sollte generell nie so weit ausgelastet sein, dass ein Swappen kompletter Prozesse notwendig wird, da dies die Performance wesentlich beeintraechtigt. Mit Hilfe des Programms /usr/etc/monitor (vgl. Systemverwalterhandbuch) koennen Sie feststellen, ob das System schon zum Swappen uebergegangen ist. Ist dies der Fall, so ist der Hauptspeicherausbau nicht ausreichend. Bei normaler Systemlast findet nur Paging und kein Swapping statt.

Aenderungen im Dateisystem

Der SINIX V5.22 Kern erlaubt Dateinamen bis zu einer Laenge von 255 Zeichen. Daher laufen Programme, die die Berechnungsregel: Groesse Dateiverz. = Anz. Eintraege * Groesse eines Eintrags zugrunde legen, u.U. nicht korrekt ab. Obwohl der SINIX V5.22 Kern auch Namen laenger als 14 Zeichen verarbeiten kann, wird Programmen, die im sie-Universum uebersetzt wurden, nur der Zugriff auf die ersten 14 Zeichen erlaubt. Im att-Universum simulieren einige Dienstprogramme zwar einen Indexeintrag von 16 Zeichen, ein direkter Zugriff auf die Dateiverzeichniseintraege liefert jedoch den vollstaendigen Eintrag. Wird die volle Pfadnamenlaenge von 1024 Bytes bzw. die volle Dateinamenlaenge von 255 Bytes ausgenuetzt, so koennen einige Utilities diese Pfadnamenlaenge nicht verarbeiten (z.B. rm -rf, ls, pwd, tar, cpio). Bitte beruecksichtigen Sie dies beim Vergeben von Datei- und Dateiverzeichnisnamen.

Interprozesskommunikation

Die Moeglichkeiten der Interprozesskommunikation sind mit SINIX V5.22 im att-Universum durch named pipes, shared memory, messages und semaphore gegeben. Mit dem Kommando `ipcs` erhalten Sie Informationen ueber einige der aktuellen Interprozessaktivitaeten.

Line discipline

Da die Universen unterschiedliche Grundeinstellungen fuer die Terminals (line discipline) benoetigen, muessen diese vor dem Universumwechsel u.U. mit dem Kommando `ucb stty` richtig eingestellt werden.

Folgende Einstellungen sind Standard:

Universum	Startprogramm	line discipline
ucb	/bin/sh	old
ucb	/bin/csh	new
att	/bin/sh	sys5
sie	/bin/sh	sys5

Symbolische Links

Die Implementierung der symbolischen Links zieht einige Konsequenzen nach sich. Wenn ein symbolischer Link auf einen Dateikatalog verweist, aendert sich bei entsprechender Konstellation das Verhalten des Systems beim Kommando `cd`. Dienstprogramme, die `cd` verwenden, um aus bestimmten Dateikatalogen in das zugehoerige uebergeordnete Dateiverzeichnis zurueckzukehren, liefern beim Ueberqueren von symbolischen Links unerwartete Resultate.

Der Verweiszaehler einer Datei ist unabhaengig von den symbolischen Links, die fuer diese Datei eingerichtet wurden. Damit ist es moeglich, das Ziel eines symbolischen Links zu loeschen. Das hat totalen Datenverlust zur Folge, und es existieren Namen von Dateien in Dateikatalogen, deren Inode und Dateiinhalt es nicht mehr gibt.

Beachten Sie, dass `/vmunix` ein symbolischer Link auf `/dynix` ist, d.h. ein Loeschen der Datei `/dynix` fuehrt zum Verlust des Betriebssystemkerns.

Ein uebersetztes Programm kann in jedem Universum ablaufen. Das Ablaufverhalten kann jedoch in den Universen unterschiedlich sein.

Ablaufumgebung SINIX V2.1 - sie-Universum

Mit SINIX V5.22 steht das sie-Universum nur noch als reine Ablaufumgebung zur Verfuegung. Eine zu SINIX V2.1 kompatible Entwicklungsumgebung ist damit unter SINIX V5.22 nicht mehr vorhanden.

Systemdatei /etc/group und Umgebungsvariable TARMAPGID

Durch das Softwareprodukt DFS V1.0 und die hiermit einhergehende Moeglichkeit, mit heterogenen Netzen zu arbeiten, ist es notwendig, gegenueber den SINIX Versionen 2.1, 5.0 und 5.1 die Gruppennummern (GID) von other und daemon zu tauschen.

Gruppe	V2.1/V5.1	V5.22
other	1	10
daemon	10	1

Um bei den Softwareprodukten des sie-Universums zu gewaehrleisten, dass die GID's richtig installiert werden, wurde eine Aenderung im tar des sie- und des Systemverwalter-Universums vorgenommen und die Umgebungsvariable TARMAPGID implementiert. TARMAPGID wird in der Datei /etc/superinstall gesetzt: TARMAPGID=10=daemon,1=other. Beim Einlesen von Daten mit dem sie- oder ucb-tar werden Dateien, die auf dem Datentraeger zur GID 10 gehoeren, auf der Platte der Gruppe daemon zugeordnet und umgekehrt.

Um neue Produkte mit den bereits richtigen GID's von alten Produkten zu unterscheiden, muss bei neuen Softwareprodukten zusaetzlich zur Datei install die Datei install.conf existieren. In der Datei install.conf kann man mit UNIVERSE=... das Universum festlegen, in dem die Software installiert werden soll. Ausserdem koennen zusaetzlich noch die Variablen PRODUCT, VERSION, RELEASE und UPDATE versorgt werden; hierdurch werden diese Informationen automatisch in die Datei /etc/versionlog eingetragen.

```
Beispiel: PRODUCT="SINIX"
          VERSION="V5.21A10"
          UPDATE="N3"
          RELEASE="11.11.89"
```

Systemdatei /etc/gettytab

Die Datei /etc/gettytab wurde um die Eintraege N fuer das 8-Bit-Terminal 97801-4* und v fuer das Fremdterminal vt220 erweitert.

Systemdatei /usr/crash/minfree

In der Datei minfree ist hinterlegt, wieviel Speicherplatz auf dem Plattenbereich, auf dem sich das Dateiverzeichnis /usr/crash befindet, mindestens zur Verfügung stehen muss, damit /etc/savecore einen vmcore nach /usr/crash sichert. Die Datei ist vorbelegt mit 1000 (Fragmenten). Der Wert sollte abhängig von der Größe des Hauptspeichers hochgesetzt werden.

Das Dateiverzeichnis crash muss nicht zwangsweise im Dateisystem /usr eingetragen sein, sondern kann sich auch in beliebigen anderen Dateisystemen befinden; sie müssen lediglich darauf achten, dass es sich direkt unterhalb des Dateisystems befindet (z.B. /usr1/crash, aber nicht /usr1/xxx/crash). In diesem Fall müssen Sie nur die Umgebungsvariable CRASH_FS in der Datei /etc/rc entsprechend versorgen (z.B. CRASH_FS=/usr1).

Geänderte Gerätedateien für Terminals und Drucker

Die Major- und Minornummern für die Terminals 97801 und 97808 sind gleich, d.h. eine Minornummer < 100 bedeutet keinen geschützten Terminalbetrieb mehr. Die Minornummern devno und devno + 100 bezeichnen in Zukunft verschiedene Schnittstellen. Die Minornummern werden mit 0 beginnend der Reihe nach durchnummeriert. Die Majornummer für alle Terminal- und Druckerschnittstellen an E/A-Prozessoren ist 38; z.B.:

```
crw--w--w- 1 root root 38, 3 Apr 19 07:06 /dev/tty003
```

Vorsicht: Mit SINIX V5.22 wurden die Nummern für Terminals und Drucker auf 3 Stellen erweitert! (s.o.)

Anwendungen, die die Terminalschnittstelle in den raw-Modus stellen, müssen mit Datenverlusten rechnen. Prinzipiell sollten Anwendungen nie den raw-Modus einstellen, sondern immer nur den cbreak-Modus.

Gerätedateien für Terminals und Drucker an SIM-Baugruppen

Die Majornummer für Terminals und Drucker an SIM-Baugruppen ist immer 33. Die Minornummer berechnet sich aus der 3-stelligen Nummer des Gerätedateinamens minus 200!

Beispiel: tty280 hat die Majornr. 33 und die Minornr. 80

Geraetedateien fuer Pseudo-Terminals

Bei der Systeminstallation werden standardmaessig 32 Pseudo-Terminals eingerichtet. Die uebrigen Pseudo-Terminals koennen Sie als Systemverwalter ueber das neue Bediensystem unter den Punkten "Konfigurierung - Pseudo-Terminals" einrichten.

Es koennen maximal 2046 Pseudo-Terminals eingerichtet werden.

Nach dem Einrichten neuer Pseudo-Terminals sollten sie ebenfalls die Prozedur /etc/devreorg im Single-User-Modus ablaufen lassen!

Pseudo-Terminals werden sowohl fuer LAN als auch fuer ALPHA-COLLAGE benoetigt.

Sicherung

Fuer Datensicherungen und Ruecksicherungen stehen die Programme dump und restore im Expertenmodus zur Verfuegung. Diese Programme ermoeglichen auch eine Differenzsicherung (dump) und eine gezielte Ruecksicherung einzelner Dateien oder Dateiverzeichnisse (restore). Es kann auch das Kommando tar mit Angabe der entsprechenden Geraetedatei verwendet werden.

Eine komfortable menuegefuehrte (oder kommandogesteuerte) logische Datensicherung wird mit dem Softwareprodukt DSX angeboten.

Native Language Support (NLS)

NLS liefert die Grundlage fuer

- die Entwicklung internationalisierter Programme, die gleichzeitig in verschiedenen Sprachen ablaufen koennen.
- die Unterstuetzung von 8-bit-Zeichensaetzen
- die Bekanntgabe der gewuenschten Sprache (Variable LANG).

Die Programmentwicklung mit NLS wird nur im att-Universum unterstuetzt. Internationalisierte Programme koennen jedoch auch im ucb- bzw. im sie-Universum ablaufen.

Fuer die Wirksamkeit der NLS-Funktionen sind die Environment-Variablen NLSPATH und INTLINFO massgebend. Diese sind im att-Universum voreingestellt.

Die Sprache, in der internationalisierte Programme ablaufen, ist abhaengig von der Umgebungsvariablen LANG. Diese Variable wird durch das Bediensystem eingestellt fuer die Universen ucb und att bzw. im sie-Universum auf den Wert der entsprechenden Landessprache gesetzt, in der das sie-Universum installiert wurde.

Die NLS-Kommandos findstr und insertmsg entfallen in SINIX V5.22, da sie nicht dem Standard XPG3 entsprechen.

2.5 Entfallene Funktionen

vgl. 2.7.

Mit SINIX V5.22 wird im sie-Universum keine Entwicklungsumgebung mehr zur Verfügung gestellt.

2.6 Inkompatibilitaeten

Die im sie-Universum erstellten Anwendungen koennen unter Beachtung bestimmter Einschränkungen (siehe unten) im Binaerformat auf SINIX V5.22 portiert werden.

Von der Binaerkompatibilitaet ausgenommen sind Programme, die bestimmte Annahmen ueber die zugrundeliegende Implementierung treffen. Hierzu zaehlen Programme, die beim Systemstart aufgerufen werden, sowie Dienstprogramme, die auf Betriebssystemtabellen oder die interne Struktur des Dateisystems zugreifen.

Softwareprodukte, die Geraetedateien fuer Drucker, Plotter, Bildschirme usw. anlegen, duerfen nicht unbedenklich von alten Sicherungen uebernommen werden, da sich evtl. (beim Uebergang von SINIX V5.1x auf SINIX V5.2x) die Major- und Minornummern geaendert haben. Zudem haben sich die Namen der Geraetedateien fuer Drucker und Terminals geaendert (tty004 ist entweder ein Bildschirm oder ein Drucker an der fuenften Schnittstelle).

Programme, die den Dateibaum rekursiv durchlaufen, koennen aufgrund symbolischer Links unerwartete Resultate liefern.

Einige Systemdateien haben sich im Format geaendert.

Das Dateiverzeichnisformat hat sich geaendert.

Dateien, die mit .# beginnen, werden nicht mehr - wie in SINIX V.21 - automatisch ueber die Shell-Prozeduren daily, weekly oder monthly im Dateiverzeichnis /usr/adm geloescht.

2.7 Einschränkungen

2.7.1 Online-Dokumentation

SINIX V5.22 beinhaltet keine Online-Dokumentation. Diese Dokumentation wird auf einer separaten Magnetbandkassette zur Verfügung gestellt und kann bei der für Sie zuständigen Zweigniederlassung bezogen werden. Da nach der Installation des Grundsystems die Manualseiten nicht zur Verfügung stehen, können folgende Funktionen nicht ausgeführt werden:

man Beschreibung zu einem Kommando anzeigen

catman Erzeugen der vorformatierten Ausgabe der Online-Kommandobeschreibung aus Eingabe-Dateien im nroff-Format

2.7.2 Einsatz von zusätzlichen Softwareprodukten

Für die folgenden Softwareprodukte ist unter SINIX V5.22 mindestens der nachfolgend aufgeführte Versionsstand einzusetzen:

ACX	V3.0B
ANIM85	V1.0A
BASIX-COMP	V3.0A
BASS	V2.0A10
CCP-LAN1	V3.0A20
CCP-LAN2	V1.5
CCP-STA1	V2.1B
CCP-WAN1/2/5/6	V1.5A
CMX	V2.1A
COB85	V1.0A10
COLTK	V1.12
C++	V2.0A
DBX	V1.1B
DFS	V1.0A20
EMDS	V2.1B
EMX28	V1.2A
ESQL-C	V2.1A30
ESQL-COBOL	V2.1A30
FORTRAN	V1.1A10
FT-SINIX	V2.2
HIT	V4.0A
HIT-COL	V3.1A
INFORMIX-D	V2.1A30
INF-NET	V1.1A
INFORMIX-R	V2.1A30
INF-TURBO	V1.1A10
INF-4GL-D	V1.1A30
INF-4GL-R	V1.1A30
LISP	V1.1A10
PASCAL-XT	V1.2A

PROLOG	V2.0A	
REMOS	V3.0A20	
REMOS2	V2.1A20	*)
SICHART	V2.0B	
SIDRAW	V2.0B	
SIPLAN	V2.0B	
TRANSIT	V2.1A	
TSX29	V1.2A	
UX-BASIC	V3.0A	
XTI	V1.0A	

*) Lieferfreigabe Mai 1990

OCIS-DESK V1.0A10 ist unter SINIX V5.22 nicht freigegeben!

Das auf Sonderfreigabe erhaeltliche CCP-STA3 und CCP-STA4 kann nur eingesetzt werden, wenn kein NCR-Hostadapter gesteckt ist (vgl. Kapitel 5.11).

2.7.3 Einsatz des Dienstprogramms /etc/addbad

Das Dienstprogramm /etc/addbad zum Auslagern defekter Spuren auf Festplatten darf auf keinen Fall eingesetzt werden, da es dabei zu Systemabstuerzen und Datenverlust auf der betreffenden Festplatte kommen kann. Bei auftretenden Hardwarefehlern auf einer Ihrer Festplatten (Meldung: HARD ERROR ...) wenden Sie sich bitte umgehend an den zustaeendigen Wartungsdienst und nehmen keine eigenstaendigen Reparaturen vor.

2.7.4 Kommando sendmail

Mit dem Kommando sendmail duerfen keine Dateien groesser als 100 KB verschickt werden!

2.7.5 NLS-Funktionsaufrufe

Folgende im Manual Internationalisation (Bestellnummer U5098-J-Z95-1-7600) beschriebene Schnittstellen sind nicht freigegeben und koennen sich jederzeit aendern:

```
form_lang(3x)
i_asctime(3x)
i_cnvtyp(3x)
i_end(3x)
i_getstr(3x)
i_init(3x)
i_ldcncv(3x)
i_ldcol(3x)
```

```

i_ldprp(3x)
i_ldstr(3x)
i_perror(3x)
i_prp(3x)
i_string(3x)

```

2.7.6 Modemfaehige Schnittstelle V24

Die im Schnittstellen-Handbuch beschriebene SEAAC-Baugruppe mit einer modemfaehigen Schnittstelle V24 wird erst mit SINIX V5.23 unterstuetzt. Ebenso ist die Schnittstelle V24 an einer SIM-Baugruppe derzeit nicht modemfaehig.

2.7.7 Bedienung des Magnetbandkassettenlaufwerks Video-8 mittels mt-Kommando

Die Funktionen `seod` und `eod` des Dienstprogramms `mt` zum Positionieren auf das Ende der Daten funktioniert in SINIX V5.22 nicht; ebenso die Funktion `status`, die den aktuellen Status des Geraetes abfragt.

Hinweis zum Zurueckpositionieren des Bandes:
nach einem Rueckwaertspositioniervorgang mittels `mt ... bsf` steht der Schreib-/Lese-Kopf vor der Filemark, daher muss anschliessend die Filemark uebersprungen werden (mit `mt ... fsf!`)

Beispiel:

Der Schreib-/Lesekopf steht am Anfang des fuenften Bandarchives (a4) und es soll auf das vierte Bandarchiv (a3) positioniert werden:

```

1   2   3   4   5   6   7
a0  a1  a2  a3  a4  a5  a6  . . . .

```

zu diesem Zweck positionieren Sie nun um 2 Bandmarken zurueck:

```
ucb mt -f /dev/exa8 bsf 2
```

anschliessend steht der Schreib-/Lesekopf am Ende des dritten Bandarchives (a2)!

Um nun das vierte Bandarchiv (a3) bearbeiten zu koennen, muss der Schreib-/Lesekopf am Anfang des Archivs positioniert sein. Dies erreicht man mit folgendem Kommando:

```
ucb mt -f /dev/exa8 fsf 1
```

2.8 Verhalten im Fehlerfall

Im Fehlerfall werden zu Diagnosezwecken Fehler-Unterlagen benoetigt. Diese schicken Sie bitte zusammen mit der Fehlermeldung an Ihre zustaeundige Zweigniederlassung.

Zu den Fehlerunterlagen gehoeren eine genaue Beschreibung der Fehlersituation und Angaben, ob und wie der Fehler reproduzierbar ist. Grundsaeztlich wird fuer die Fehlerdiagnose zusaetzlich ein Hauptspeicherabzug benoetigt (vgl. "Ausfall des Systems").

Plattendefekte

Sollten waehrend des Betriebs von SINIX V5.22 Plattendefekte auftreten, so verstaendigen Sie bitte den zustaeundigen Wartungsdienst.

Auf keinen Fall sollten Sie selbst die im Systemverwalterhandbuch beschriebenen Dienstprogramme zur Behandlung defekter Spuren etc. verwenden, da mit diesen Programmen bei unsachgemaesser Bedienung mehr zerstoert als repariert wird (vgl. auch Kapitel 2.7.3).

Fehlerhafte Baugruppen

Falls waehrend des Betriebs von SINIX V5.22 Fehler auftreten, die defekte Baugruppen vermuten lassen, so verstaendigen Sie bitte den zustaeundigen Wartungsdienst.

Ein Dateisystem ist voll

Ueberpruefen Sie mit dem df-Kommando, welches Dateisystem voll ist.

Ist das usr-Dateisystem voll, so sollten Sie die Dateiverzeichnisse /usr/adm und /usr/crash auf grosse Dateien durchsuchen, die geloescht werden koennen. Im Dateiverzeichnis /usr/adm wachsen folgende Dateien staendig an und sollten von Zeit zu Zeit in ihrer Groesse reduziert werden (mit dem Kommando "> dateiname").

/usr/adm/lastlog	/usr/adm/sus
/usr/adm/messages.*	/usr/adm/wtmp.*
/usr/adm/shutdownlog	

Ist das root-Dateiverzeichnis voll, so ueberpruefen Sie bitte, ob im root-Dateisystem Dateien stehen, die nicht zum Grundsystem gehoeren. Loeschen Sie diese gegebenenfalls.

Ausfall des Systems

In allen Fehlerfaellen schreiben Sie bitte die Systemmeldungen ab, die auf der Konsole ausgegeben werden. Sie koennen eine grosse Hilfe bei der Fehlerdiagnose sein. Z.T. werden diese Meldungen in den Dateien /usr/adm/messages und /usr/adm/shutdownlog mitprotokolliert.

Nach einem Systemabsturz wird automatisch ein kompletter Arbeitsspeicherabzug vom Standalone-Programm dump in den Swap-Bereich geschrieben, vorausgesetzt die Bootflags wurden von Ihnen nicht veraendert und die Taste AUTOBOOT an der Systemeinheit ist gedruickt.

Wollen Sie ohne einen eigentlichen Systemabsturz gezielt einen Systemdump erzeugen, so betaeligen Sie im laufenden Betrieb die RESET-Taste, wobei Sie dann darauf achten muesen, dass die AUTOBOOT-Taste nicht gedruickt ist. Das System befindet sich dann im Einschaltmonitor, in dem Sie jetzt mit dem Kommando b80 einen Systemdump erzeugen koennen. Daten, die noch nicht auf der Platte stehen, gehen hierbei verloren.

Beim erneuten Hochfahren wird der Hauptspeicherabzug dann unter dem Namen vmcore.* und der aktuelle Kern unter dem Namen dynix.* im Dateiverzeichnis /usr/crash gesichert. Der Speicherauszug kann fuer die Fehlerdiagnose wichtig sein. Achten Sie darauf, dass auf der usr-Partition immer genuegend Platz (16 bis 64 MB, abhaengig vom Hauptspeicher-ausbau) fuer einen Hauptspeicherabzug vorhanden ist.

Im Fehlerfall schicken Sie bitte immer alle Dateien aus dem Dateiverzeichnis /usr/crash, die dieselbe Endnummer haben, und zusaetzlich das Programm /etc/crash (bzw. /usr-etc/crash) zusammen mit allen anderen Fehlerunterlagen an Ihre Zweigniederlassung. Achten Sie dabei bitte darauf, dass alle Dateien mit relativen Pfadnamen gesichert werden.

3 Software-Erweiterungen

3.1 Installation

Beim erstmaligen Hochfahren von der Festplatte in den Mehrbenutzerbetrieb wird die KEY-Diskette K.DISK angefordert. Es wird geprüfert, ob diese KEY-Diskette zur Inbetriebnahme des Systems berechtigt; vgl. auch 2.3.

3.2 Bediensystem

SINIX V5.22 wird mit einem neuen Bediensystem auf Basis von COLLAGE V3.0B und dem Dialogmanager V2.0A ausgeliefert. Es unterstützt sowohl den Grafik-Modus von 97808-Terminals als auch den alphanumerischen Modus der 97801-Terminals. Das COLLAGE-Bediensystem bietet die volle Funktionalität von COLLAGE, erweitert um Funktionen und Programme, die mit denen des bisherigen Standardmenuesystems auf Basis des MES vergleichbar sind.

Als Systemverwalter mit der Kennung "admin" erhalten Sie als Oberfläche das COLLAGE-Bediensystem. Systemverwalterfunktionen führen Sie abhängig von Ihrem Terminaltyp unter COLLAGE oder unter Alpha-COLLAGE aus. Unter Alpha-COLLAGE sollten Installationen von Software-Produkten ausschließlich im Full-Screen-Modus durchgeführt werden, da die meisten Installationsprozeduren die 25. Bildschirmzeile benutzen, die nur im Full-Screen-Modus sichtbar ist.

Zum Arbeiten mit dem Bediensystem auf Alpha-Terminals benutzen Sie bitte den mitgelieferten Tastaturstreifen zu COLLAGE V3.0B (Bestellnr.: U5134-J-Z98-1).

Anwender können wahlweise mit dem COLLAGE-Bediensystem, dem Menuesystem auf Basis MES oder auf Shellebene arbeiten. Die Anzahl der Anwender, die unter COLLAGE bzw. unter Alpha-COLLAGE arbeiten können, ist vom Hauptspeicherausbau und der Größe des Swap-Bereichs abhängig; maximal dürfen die Hälfte aller Terminals mit Alpha-COLLAGE betrieben werden.

Das "Konfigurieren des lokalen Systems" wurde den Gegebenheiten der Version SINIX V5.22 angepasst. Neu ist die Konfigurierung der SIM-Boards und TAK's. Über die Geräteauswahl "97801-8bit" können 8-Bit-Terminals konfiguriert werden und über die Geräteauswahl "Spezialgeraet" ist es jetzt möglich, sowohl für Schnittstellen V24 als auch für Schnittstellen SS97 Nicht-Standard-Geräte einzutragen.

Das Terminal 97801-480 ist per Setup-Menü einstellbar auf 7-Bit oder 8-Bit-Modus. Man kann über das Bediensystem die Schnittstelle generell auf 8 Bit einstellen und je nach Anwendung per Setup-Menü die 7-Bit- oder 8-Bit-Emulation auswählen.

Die Geraete zum Betrieb von ASECO (CKT, CKL, SM2 und HKT) duerfen nicht an SIM-Baugruppen konfiguriert werden.

Achtung: Die Benutzerklasse der Anwender kann von menu auf collage, aber nicht von collage auf menu geaendert werden.

Das COLLAGE-Bediensystem benutzt Lock-Dateien, um den exklusiven Aufruf einzelner Anwendungen sicherzustellen. Lock-Dateien werden normalerweise von der Anwendung bei Beendigung geloescht. Bei einem Programmabbruch oder nach einem Systemabsturz kann es noetig werden, dass die Lock-Dateien von Hand geloescht werden muessen. Betroffen sind ggf. die Dateien /tmp/Konf, /tmp/Login, /tmp/Admprint, /tmp/Prntkonf, /tmp/Admprint.<uid>.

Die Anwendung Post setzt auf dem SINIX-Kommando /usr/bin/mailx auf.

Fuer die mit MES entwickelten Menueanwendungen steht im COLLAGE-Bediensystem eine Ablaufumgebung zur Verfuegung. Software-Produkte, wie z.B. LAN/REMOS, haben die Moeglichkeit, eigene Menues in das COLLAGE-Bediensystem einzuhaengen, die dann den Benutzern ueber die Menueleiste zur Auswahl angeboten werden. Die geaenderte Menueleiste wird jedoch erst nach einem erneuten Start des Bediensystems angezeigt.

Zum Anschluss eines Software-Produktes an das COLLAGE-Bediensystem sind folgende Schritte noetig:

1. Anlegen einer Datei <helpmsgfile> mit Hilfsinformationen in der Form
 - Zeile 1 n
 - Zeile 2 n
 - Zeile n.
 Die Datei darf nicht mehr als 250 Zeichen enthalten.
2. Anlegen einer Datei <itemfile>. Diese Datei enthaelt den Eintrag, der in die Menuedatei uebernommen werden soll.
3. Integration des Aufrufs "Addtomenu" in die Installationsprozedur:


```
Addtomenu <menufilename><helpmsgfile><itemfile> <lang>
```

 Addtomenu haengt den Inhalt von <itemfile> an die Datei <menufilename>.mnu an. Der Inhalt von <helpmsgfile> wird als Message in den Katalog <menufilename>.cat aufgenommen.

vgl. Handbuch: "COLLAGE-Bediensystem".

3.3 Virtuelles Plattensubsystem (VPSS)

Um Datenverlust bei Plattenproblemen zu vermeiden bzw. um in einem solchen Fall ohne grosse Zeitverzoegerung weiterarbeiten zu koennen wird ab SINIX V5.22 ein virtuelles Plattensubsystem (VPSS/Spiegelplattensystem) unterstuetzt. Damit ist ueber Pseudogeratetreiber fuer den Benutzer transparent die automatische Duplizierung der Daten einer Partition moeglich. Die Duplizierung der Daten erfolgt von einer primaeeren auf eine oder mehrere sekundaere Partitionen.

Folgende Hinweise und Einschraenkungen sind zu beachten:

- Das VPSS kann nur fuer raw-devices eingesetzt werden, also nicht fuer Plattenbereiche, die ein Dateisystem enthalten. Im Wesentlichen kann VPSS somit fuer Informix-Turbo eingesetzt werden.
- Falls Primary- und Secondary-Partition nicht gleich gross sind wird die kleinere der beiden Partition-Groessen fuer den nutzbaren Bereich verwendet.
- Eine c-Partition darf nicht gespiegelt werden, da sie die Fehlstellenliste enthaelt!
- Spiegelung einer a-Partition:
Secondary Partition darf nur eine a-Partition sein!
Die Anwendung muss dabei den ersten Block der Partition ueberspringen (Offset von 512 Bytes), da der Block 0 mit den Platteninformationen nicht ueberschrieben werden darf. Dieser Offset wird von VPSS nicht beruecksichtigt!
- Wuenschenswerte Konfiguration fuer Primary- und Secondary-Partition:
 - unterschiedliche Platten
 - unterschiedliche Controller
 - Unterschiedlicher Multibus-I

3.4 Meldungsdateien fuer internationalisierte Kommandos

SINIX V5.22 enthaelt standardmaessig sowohl die deutschen als auch die englischen Meldungsdateien fuer die internationalisierten Kommandos.

Fuer alle anderen Sprachvarianten sind die entsprechenden Meldungsdateien als eigene Liefereinheit zu bestellen:

Message-Files

SINIX-SPR-<Landeskennzeichen>

3.5 Kommandos

Mit SINIX Version 5.22 wurden weitere Kommandos auf der Basis von NLS internationalisiert. Die Meldungsdateien dieser sprachabhaengigen Kommandos sind im Dateiverzeichnis /usr/.lib/nls/msg hinterlegt. Beachten Sie, dass es von dem Inhalt der Variablen LANG abhaengig ist, welche Ausgaben die internationalisierten Programme (z.B. mail, sort, tr, ced) liefern.

/etc/dump

Aus aktuellem Anlass weisen wir auf folgenden Sachverhalt hin: Bitte beachten Sie, dass Sie sich bei unsachgemaessem Aufruf des Kommandos dump den Inhalt ganzer Partitions loeschen koennen. Die zu sichernde Partition darf auf keinen Fall als <argument> zum Schalter f angegeben werden. In diesem Fall wird diese Partition als Sicherungsmedium verwendet; d.h. der Inhalt wird von der Ausgabe des dump ueberschrieben. Achtung: Wenn die Geraetedatei, von der gelesen wird, identisch ist mit der Geraetedatei, auf die geschrieben wird, werden Bereiche ueberschrieben, die noch nicht gelesen wurden. Damit ist einerseits das Dateisystem zerstuert und andererseits die dump-Sicherung unbrauchbar.

/etc/restore

Das Kommando restore bietet mit der Option -a die Moeglichkeit, ein anderes Dateiverzeichnis als /tmp fuer das Anlegen temporaerer Dateien festzulegen.

/usr/bin/tar

Mit SINIX V5.22 wird fuer das xopen-Universum ein neuer tar ausgeliefert. Datentraeger, die mit dem "att tar" der Version V5.22 (XPG3-tar) beschrieben werden, koennen mit dem "att tar" bis einschliesslich V5.21 und dem "ucb tar" nicht mehr gelesen werden. Datentraeger, die im alten Format beschrieben wurden, koennen weiterhin gelesen werden. Der "att tar" (V5.21, XPG2-tar) erhaelt den Namen otar. Bei Sicherungen mit dem "att tar" der Version V5.22 werden "symbolic" und "conditional symbolic" Links aufgeloeset. Dies fuehrt zur Mehrfachsicherung. Der "att tar" (V5.22) kann zwar Sicherungen lesen, die sich ueber mehrere Magnetbaender, Kassetten oder Disketten erstrecken, kann aber derartige Sicherungen nicht schreiben. Die Verarbeitung mehrerer Archive auf einem Magnetband oder einer Kassette ist nicht moeglich. Wird ein Archiv als nicht XPG3-kompatibel erkannt, erfolgt ein "rewind", anschliessend wird der "att tar" mit dem "ucb tar" ueberladen.

`/bin/cpio`

Die Anforderung eines Folgemediums funktioniert nur fuer "nicht puffernde" Treiber (z.B. rmt0, rmt232).

`/usr/etc/cptape` - Kopieren von Magnetbandkassetten

Mit der Prozedur `/usr/etc/cptape` wird Ihnen ein Shellscript zum Kopieren von Magnetbandstreamerkassetten zur Verfuegung gestellt. Bitte verwenden Sie diese Prozedur um sich eine Sicherungskopie Ihres Installationsbandes anzulegen.

Die Prozedur kopiert keine Video-8 Kassetten!

`/etc/devreorg` - Konfigurierung zusaetzlicher Terminals

Bei der Konfigurierung zusaetzlicher Terminals werden die zugehoerigen Geraetedateien physikalisch ans Ende des Dateiverzeichnisses `/dev` angehaengt. Dies fuehrt zu erheblichen Performance-Verlusten beim Arbeiten mit diesen Terminals. Um diese Performance-Verluste zu verhindern, steht die Prozedur `/etc/devreorg` zur Verfuegung. Mit Hilfe dieser Prozedur wird das gesamte Dateiverzeichnis `/dev` dahingehend reorganisiert, dass alle Geraetedateien fuer lokale Terminals physikalisch am Anfang stehen, gefolgt von den virtuellen und den Pseudo-Terminals.

Vorsicht: Die Prozedur `/etc/devreorg` muss im Single-User-Modus aufgerufen werden!

Bitte entfernen Sie einen eventuell vorhandenen symbolischen Link von `/tmp` in eine andere Partition, da sonst die Prozedur nicht ordnungsgemaess ablaufen kann (innerhalb der Prozedur werden alle Dateisysteme abgehaengt!)

`/bin/ps`

Beim Kommando `ps` hat sich mit XPG3 die Bedeutung verschiedener Optionen geaendert.

- a Die Prozessgruppenfuehrer werden nicht angezeigt.
- l Das Ausgabeformat hat sich mit XPG3 geaendert.
- t Hier koennen jetzt mehrere Terminalnummern angegeben werden.
- e Auflisten aller Prozesse im System.
- u Auflisten aller Prozesse der Benutzer, die nach dem Schalter `-u` angegeben sind: `<ps -u uidlist>`.
- g Auflisten aller Prozesse der Prozessgruppenfuehrer, die nach dem Schalter `-g` angegeben sind: `<ps -g grplist>`.
- d Auflisten aller Prozesse ausser den Prozessgruppenfuehrern
- p Auflisten aller Prozesse, deren PID nach dem Schalter `-p` angegeben ist: `<ps -p proclist>`.

Die Schalter -c, -s, -v, -w, -x und entfallen mit XPG3.

/etc/pstat

Bei Angabe des Schalters -T kann man der Ausgabe nicht mehr entnehmen, wieviele Inodes aktuell gebraucht werden. Ursache ist die geaenderte Verwaltung der Inodes. Die Inodes werden in einem Cache verwaltet; auch Inodes, die nicht mehr gebraucht werden, bleiben in dieser Inodetabelle stehen. Wird ein Inode gebraucht, der bereits in der Inodetabelle vorhanden ist, so kann auf diesen Inode sehr performant zugegriffen werden. Muss ein neuer Inode in die Inodetabelle geladen werden, so wird ein Inode der Tabelle ueberschrieben. Dies ist der Inode, auf den die laengste Zeit nicht zugegriffen wurde (LRU-Cache).

/usr/bin/deact_ttys und /usr/bin/act_ttys

Wenn die Leitung vom Rechner zu einem SIM-Board unterbrochen ist, erscheint fuer jedes angeschlossene Terminal zyklisch eine Fehlermeldung an der Konsole. Um dies zu verhindern, wird dem Systemverwalter die Moeglichkeit geboten, Terminals zu deaktivieren und zu aktivieren.

Syntax: deact_ttys ttynnn
act_ttys ttynnn

Als weiteres Argument akzeptieren beide Programme den Schalter -l, der bewirkt, dass eine sortierte Liste aller aktiven bzw. inaktiven Terminals an SIM-Boards ausgegeben wird. Gueltige tty-Nummern fuer Terminals an SIM-Boards sind: 200 - 455.

/etc/maxusers

Das Kommando maxusers gibt den aktuellen Wert der Variablen MAXUSER aus, sofern es ohne Argument aufgerufen wird. Zum Aendern der Variablen MAXUSER kann der gewuenschte neue Wert als Argument bei /etc/maxusers angegeben werden (vgl. Kapitel 2.4).

/bin/uname

Die Ausgaben verschiedener Optionen des Kommandos uname haben sich geaendert.

Der Schalter -r liefert den Release des Betriebssystems. Der Schalter -v liefert den Stand innerhalb eines Releases.

/etc/keyload

Das Kommando keyload ist erweitert worden. Es koennen jetzt auch Tastaturtabellen fuer 8-Bit faehige Terminals 97801-480 geladen werden. Wollen Sie die Tastaturbelegung eines Terminals von einem anderen Terminal aus laden, so muessen Sie den Terminaltyp als Parameter beim keyload-Aufruf angeben und die Terminalschnittstelle richtig einstellen. Wird keyload an dem Terminal aufgerufen, dessen Tastaturtabelle geladen werden soll, so werden die Leitungsparameter durch den keyload selbst entsprechend eingestellt.

Aufrufsyntax:

```
/etc/keyload -t type tty keytable
```

type spezifiziert den Terminaltyp und kann folgende Werte annehmen:

```
new Tabelle fuer 97801 (7-Bit-Terminal)
7new Tabelle fuer 97801-480 (7-Bit-Emulation)
8bin Tabelle fuer 97801-480 (8-Bit-Emulation)
```

Beispiel:

Im att-Universum wird die Terminalschnittstelle richtig eingestellt und fuer tty003 die franzoesische Tastaturtabelle geladen. Anschliessend wird die Terminalschnittstelle wieder in den Ursprungszustand gebracht. Es wird vorausgesetzt, dass unter der Linediscipline sys5 gearbeitet wird.

```
att stty -opost < /dev/tty003
/etc/keyload -t 8bin tty003 franz
att stty opost < /dev/tty003
```

/bin/tr

Das Kommando tr wurde an XPG3 angepasst und ist jetzt in der Lage Multibyte-Buchstaben zu verarbeiten. Hieraus ergibt sich, dass beim Lesen von einem zeichenorientierten Geraet (z.B. ein Terminal) ein Zeichen moeglicherweise das erste Zeichen eines Multibyte-Buchstaben sein kann. tr wartet dann auf weitere Zeichen (maximal 3) oder das Zeichen EOF, bevor seine Ausgabe erfolgt. Dieser Effekt tritt nur auf, wenn in der internationalisierten Datenbasis Multibyte-Buchstaben definiert sind.

Bereits mit SINIX V5.21 wurde das Kommando tr internationalisiert. Der tr benutzt internationalisierte regulaere Ausdruecke; diese Ausdruecke enthalten keine Elemente, die eine Umsetzung von Gross- in Kleinbuchstaben ausdruecken koennen. Daher kann der tr nur dann Grossbuchstaben korrekt in Kleinbuchstaben umsetzen, wenn die Datenbasis bestimmte

Bedingungen erfuehlt (z.B. En_US.ASCII). Die meisten Sprachen koennen diese Bedingungen jedoch nicht erfuehlen. Eine sprachunabhaengige Umsetzung von Gross- in Kleinbuchstaben koennen Sie mit folgendem C-Programm erzielen:

```
include <stdio.h>
include <locale.h>
include <ctype.h>
main()

    int c;
    setlocale(LC_ALL,"");
    while ((c=getchar()) != EOF)
        putchar(tolower(c));
    exit(0);
```

Wenn Sie mit der Datenbasis De_DE.646 arbeiten, koennen Sie zur Umsetzung folgende Syntax verwenden:

```
tr ' :upper: ' ' :lower: '
```

3.6 Systemdateien

/etc/herald/9780?* und /etc/herald/vt200

Mit SINIX V5.22 wird ein neuer Herald ausgeliefert, der das XPG3-Logo enthaelt. Die Links fuer die einzelnen Bildschirme werden jeweils bei der Konfigurierung eingerichtet. Damit ist es moeglich, den Herald auch auf vt200-Terminals auszugeben.

/etc/versionlog

In der Datei versionlog ist hinterlegt, welche Software seit der letzten Betriebssystem-Installation auf Ihr System gebracht wurde.

```
/usr/admin/.products/SINIX
/usr/admin/.products/SINIX_V2.1
```

In dem Dateiverzeichnis /usr/admin/.products wurden Eintraege fuer SINIX und fuer SINIX_V2.1 aufgenommen. Das Kommando /etc/sysname meldet weiterhin "Version: V5.2".

3.7 Freigabemitteilung auf Diskette

Ab SINIX V5.22 ist die Freigabemitteilung als ASCII-Datei auf einer Diskette (README) Bestandteil des Lieferumfangs des Betriebssystems.

Die Diskette README kann ueber die Prozedur /etc/superinstall installiert werden. Anschliessend an die Installation ist im Dateiverzeichnis /usr/readme/sinix die Datei fgm00 vorhanden, die die Freigabemitteilung im ASCII-Format ohne Steuerzeichen enthaelt.

4 Das Spoolsystem

4.1 /etc/daemon

Es muessen immer zwei daemon Prozesse aktiv sein. Stuerzt der Sohnprozess ab, erfolgt ein automatisches Neuladen des daemons sofern der Vaterprozess noch laeuft. Das Neuladen wird bis zu 10 mal versucht. Eine regulaere Beendigung des daemon ist nur mit dem Kommando `lpr -dg` oder dem Signal SIGTERM moeglich. Wird dem daemon das letzte noch laufende Backend genommen, so beendet sich der daemon nicht mehr automatisch.

4.2 Backends

Beim Absturz eines Backends wird versucht, dieses neu zu laden. Bei normaler Beendigung des Backends und bei Beendigung durch die Signale SIGTERM, SIGFPE, SIGSEGV und SIGTRAP wird ein Neuladen nicht versucht.

Eigene Backends koennen wie bisher ueber das Interface-Backend `/usr/spool/spooler/interface` angeschlossen werden. "Echte" Fremdprinterbackends, die fuer die SINIX-Versionen 2.x, 5.0, 5.1A bzw. 5.2A entwickelt wurden, funktionieren mit dem Spoolsystem der V5.22 nicht mehr. Backends, die fuer V5.1B und V5.21 entwickelt wurden, koennen auch auf V5.22 eingesetzt werden. Mit Hilfe des neuen Schnittstellenhandbuchs koennen auch Fremdprinterbackends selbst geschrieben werden.

4.3 Dateien unter `/usr/.spool/spooler`

Die Dateien `CONFIG`, `CONFIG.bin`, `POOLDAT`, `FORMTAB`, `digest`, `startup`, `drucker`, `gruppen` und die Backendprogramme stehen in SINIX V5.22 im Dateiverzeichnis `/usr/.spool/spooler`. Die Dateien `drucker` und `gruppen` werden erst vom Bediensystem erstellt und auch nur vom Bediensystem benoetigt.

Das Unterdateiverzeichnis `/usr/.spool/spooler/tmp` enthaelt die Datei `daemtrc` und die Tracedateien der Backends und des daemon.

Das Dateiverzeichnis `/usr/.spool/spooler/sp` enthaelt alle temporaeren Druckdateien, die ueber eine Pipe oder ueber die `lpr`-Schalter `+co` und `-cp` an den Spooler uebergeben werden.

Das Dateiverzeichnis `/usr/.spool/spooler/at` enthaelt fuer jeden Druckauftrag eine Datei `at.<ID_Nummer>`.

Im Dateiverzeichnis `/usr/spool/spooler/stat` sind die Statusdateien hinterlegt.

Im Dateiverzeichnis /usr/.spool/spooler/font gibt es Dateiverzeichnisse fuer die einzelnen Drucker, in denen Dateien abgelegt werden koennen, die "Ladbare Zeichengeneratoren" enthalten. Diese Dateien koennen mit dem Schalter -font=<Dateiname> in den Drucker geladen werden.

Die Datei dtype steht als Bediensystemdatei unter /usr/admin/.colface/Konf. Der Aufbau dieser Datei hat sich wie folgt geaendert:

Struktur der Eintraege (eine Zeile pro Drucker, Spaltentrenner ";")

1. Druckername (wird waehrend der Konfigurierung angezeigt)
2. Backend mit absolutem Pfadnamen (dieser wird in die CONFIG-Datei in einfachen Hochkommas eingeschlossen)
3. moegliche Schalter (bis zum Zeilenende)

Beispiel:

```
9001;/usr/spool/spooler/lp9001;-pb= -pb1 -pb2 -pb3 ...
```

Die Dateien conV11 und conV24 werden nicht mehr angelegt, da im Bediensystem an V11 und V24 alle Drucker konfiguriert werden koennen.

4.4 Aenderungen und Erweiterungen

Die Datei FORMTAB ist um drei Eintraege erweitert worden. Diese drei neuen Formulare werden nur vom HP-LaserJet-Backend unterstuetzt.

Neue Backends:

Es werden mit der SINIX V5.22 zwei neue Druckertypen unterstuetzt:
9001-3 bzw. 9001-3b und HP-LaserJet-kompatible Drucker.

Backends:

```
lp9001-3    fuer Drucker 9001-31
lp9001-3b  fuer Drucker 9001-8931

lphplj     fuer HP-LaserJet-kompatible Drucker
```

Hinweis:

Da der HP-LaserJet keine Protokollmoeglichkeiten bietet, wird der Drucker immer in den Zustand BEREIT gesetzt. Das hat zur Folge, dass Druckauftraege verloren gehen, wenn der Drucker nicht eingeschaltet ist!

Fuer die Datenuebertragung notwendige Druckereinstellungen:

- 9001 keine oder ungerade Paritaet, keine Statusmeldungen, dies entspricht 8-Bit Grafik ein
- 9011 7 Datenbits, ungerade Paritaet, keine Statusmeldungen, 9600 Baud
- 9004 9600 Baud
Die Schalterfelder an der Rueckwand des Druckers sind wie folgt einzustellen:
"BAUD RATE" 1 2 3 4
AUS EIN AUS EIN

"PROTOCOL" 1 2 3 4 5 6 7 8
AUS AUS EIN AUS AUS EIN AUS AUS.
- 9012 XON/XOFF-Protokoll, Statusmeldungen, 8 Datenbits, ungerade Paritaet, 9600 Baud
- 9013 XON/XOFF-Protokoll, Statusmeldungen, 7 Datenbits, ungerade Paritaet, 9600 Baud
- 9022 XON/XOFF-Protokoll, 8 Datenbits, keine Paritaet, 9600 Baud
- 9025 XON/XOFF-Protokoll, 8 Datenbits, keine Paritaet, 19200 Baud
- 9047 XON/XOFF-Protokoll, ungerade Paritaet, 19200 Baud
- HP-LaserJet-kompatible Drucker
XON/XOFF-Protokoll, 8 Datenbits, keine Paritaet, 19200 Baud

Es wird empfohlen, keine Drucker an der ersten Schnittstelle eines E/A-Boards anzuschliessen. Es wird vom Systemkern beim Hochfahren des Systems auf den Steckplatz 1 ein Blank geschickt. Ist nun ein Drucker an diesem Steckplatz angeschlossen, so reagiert dieser mit dem Abdruck eines Schmierzeichens, da die Baudrate nicht uebereinstimmt.

Mit SINIX V5.22 haben Sie die Moeglichkeit, in der Datei CONFIG den Standardschalter `-speed=...` anzugeben. Damit koennen Sie die Uebertragungsrateselbst festlegen (siehe auch SPOOL-Handbuch).

Geaenderte Geraetedateien fuer Drucker

Mit SINIX V5.22 wird vom Bediensystem der Name der Geraetedatei des Druckers immer mit `tty<3stellige_Nummer>` vergeben, d.h. mit `ls -l /dev` kann man nicht feststellen, welche Schnittstellen durch Drucker belegt sind.


```
crw----- 1 root    daemon 38, 2 Mar 19 07:06 /dev/tty002
```

Beispieleintrag aus der Datei CONFIG:

```
D002 '/usr/spool/spooler/lp9001' /dev/tty002 -pb= -pb1 ...
```

4.5 Inkompatibilitaeten

Zu UNIX System V Release 2 (UNIX ist eingetragenes Warenzeichen von AT&T) ist das SINIX-Spoolsystem nicht kompatibel, es existiert im att-Universum jedoch ein X/OPEN-kompatibler Spool. "Echte" Fremdrunderbackends, die fuer die SINIX-Versionen 2.x, 5.0, 5.1A bzw. 5.2A entwickelt wurden, funktionieren mit dem Spoolsystem der V5.22 nicht mehr. Backends, die fuer V5.1B und V5.21 entwickelt wurden, koennen auch auf V5.22 eingesetzt werden.

Durch die erhoehrte Anzahl der Schnittstellen mit SINIX V5.22 ist es noetig, die Druckernamen und Druckergruppennamen entsprechend zu aendern. Das hat zur Folge, dass die Nummern 3stellig statt vorher 2stellig sind. Alle Druckaufrufe (z.B. in HIT) sprechen nun Druckergruppen an, die nicht mehr existieren. Es ist deshalb zu empfehlen, zusaetzliche Druckergruppen mit den alten Namen einzurichten. Dadurch muessen nicht alle Shell-Prozeduren angepasst werden.

Beispiel:

```
G015 ( D015 ) 'DRUCKER 9022'
G15 ( D015 ) 'Zusaetzliche Druckergruppe fuer 9022'
```

Der Druckername kann nicht ueber das Bediensystem geaendert werden (im Beispiel D015).

Der Schalter -nk=... wird vom Drucker 9012 nicht mehr unterstuetzt. Um Dateien mehrfach auszudrucken, muss der Schalter -nc=... verwendet werden.

Der Spool belegt zwei Message-Queues mit den festen Werten 113 und 114. Diese Nummern duerfen daher von Anwenderprogrammen nicht belegt werden. Sind diese Nummern schon vergeben, wenn der Spool gestartet wird, so laesst sich dieser nicht laden.

4.6 Probleme mit dem SPOOL

Zu Testzwecken ist es moeglich, fuer die Druckerverwaltung (daemon) und die Backends eine TRACE-Funktion einzuschalten. Alle Zustaende der Backends bzw. des daemon werden dann in den TRACE-Dateien BTRC* bzw. TRACE protokolliert.

Die TRACE-Funktion fuer die Backends wird durch Angabe des Schalters -trace in der CONFIG-Datei eingeschaltet. Die TRACE-Funktion des daemon wird mit dem Schalter -trace beim Aufruf des daemon aktiviert. Alle erzeugten TRACE-Dateien werden im Dateiverzeichnis /usr/spool/spooler/tmp abgelegt. Im Fehlerfall finden Sie die letzten Rueckmeldungen des Druckers in der Datei /usr/spool/spooler/stat/STAT'druckername'.'pid' (siehe auch SPOOL-Handbuch).

Beispiel:

```
startup: /etc/daemon -rc -trace > tmp/daemtrc
CONFIG: D002 '/usr/spool/spooler/lp9001 -trace' ...
```

Bei schwerwiegenden Fehlern legen Sie bitte der Fehlermeldung diese Dateien auf einer Diskette bei.

4.7 Neue Schalter

/bin/lpr

Das Kommando lpr wurde um die Schalter -speed=..., -lang=... und -ds=... erweitert (siehe auch SPOOL-Handbuch).

```
-speed= Uebertragungsrate festlegen
-lang= Sprache einstellen, mit der ausgegeben wird
-ds= doppelseitiges Drucken (bei HP-LaserJet)
```

5 Hardware

5.1 Grafik-Bildschirm

In der Betriebsart "alphanumerisch" wird der Bildschirmarbeitsplatz 97801 bis auf kleine Einschränkungen emuliert. Die Darstellungsattribute (halbhell, blinkend, invers) stehen in der grafischen und alphanumerischen Betriebsart hardwareunterstützt zur Verfügung.

Folgende Escape-Sequenzen werden innerhalb von COLLAGE nicht unterstützt (vgl. Schnittstellenhandbuch):

ESC 0w	Anfordern des Schliesselschalterstatus
ESC 6u	Umstellen auf internationale Tastaturbelegung
ESC 7u	Umstellen auf deutsche Tastaturbelegung
ESC 9v	Gesamten Zeichengenerator ausgeben

5.2 8-Bit-Terminal

Ab SINIX V5.22 wird ein 8-Bit-Terminal auch als Konsole unterstützt. Hierzu ist jedoch der Firmwarestand 43A auf dem SCED-Board Voraussetzung.

5.3 Dual-CPU-Prozessor-Baugruppen

Mit SINIX V5.22 werden die Prozessoren NS32032, NS32332 und NS32532 der Firma NSC unterstützt. In den Modellen MX500-75 und MX500-85 sind standardmässig die Prozessoren NS32532 eingebaut.

Der Prozessor NS32532 ist ein 32 Bit Prozessor, der mit einer Taktfrequenz von 25 MHz betrieben wird.

Es koennen maximal 4 CPU-Baugruppen mit je 2 Prozessoren in den MX500 eingebaut werden.

5.4 E/A-Baugruppen

Es koennen maximal 16 E/A-Baugruppen am MX500 angeschlossen werden. Jede E/A-Baugruppe besitzt 6 Schnittstellen, wobei zwischen 3 Varianten gewaehlt werden kann:

- E/A-Baugruppe mit 6 x SS97
- E/A-Baugruppe mit 4 x SS97 und 2 x V24
- E/A-Baugruppe mit 6 x V24.

Generell koennen bis zu 3 AFP-Anschlusse pro E/A-Baugruppe eingebaut werden, ueber die dann durch Ergaenzung des E/A-Prozessors mit einem AFP-Anschlusszusatz mit Erweiterung drei Arbeitsplaetze, bestehend aus Bildschirmeinheit und Drucker (bzw. 2 Bildschirmen), im Inhouse-Bereich bis zu einer Entfernung von 2000 m angeschlossen werden koennen.

Bei Verwendung von mehr als 24 Bildschirmarbeitsplaetzen an einer MX500-75 bzw. MX500-85 muessen AFP-Baugruppen verwendet werden.

Mittels des SLIP-Protokolls koennen TCP/IP-Verbindungen ueber asynchrone Leitungen zwischen Rechnern hergestellt werden. Pro E/A-Baugruppe ist eine SLIP-Verbindung freigegeben. Diese kann an SS97-Schnittstellen mit einer Geschwindigkeit von 38400 Baud betrieben werden, an RS232-Schnittstellen mit 19200 Baud. Die SLIP-Verbindung ist nicht fuer den Betrieb von DFS freigegeben.

Eine Beschreibung von SLIP finden Sie im 1. Nachtrag zum REMOS-Manual V3.0A.

5.5 TACSI (Terminal Attachment Concept of SINIX)

Zusaetzlich zu den bekannten E/A-Baugruppen unterstuetzt SINIX V5.22 ein neues Terminal-Anschluss-Konzept - TACSI.

TACSI ist ein Terminal-Anschluss-Konzept in SINIX fuer den Inhouse-Bereich. Es bietet mit seiner Busstruktur die Moeglichkeit, an zwei 2-Drahtleitungen ueber vier TAK's (Terminal-Anschluss-Konzentrator) maximal 16 SINIX-Endgeraete anzuschliessen. Damit ist die Anzahl der insgesamt anschliessbaren Endgeraete (Terminals und Drucker) auf 256 begrenzt.

Vorsicht: Die Anzahl der gleichzeitig arbeitenden Benutzer ist vom Betriebssystem auf 128 festgelegt!

TACSI besteht aus den folgenden Komponenten:

- SIM (Serial Interface Multiplexer)

Diese Multibus-I-Flachbaugruppe kann systemseitig anstelle von E/A-Prozessoren installiert werden. Sie stellt zwei AFP-2-Draht Busanschluesse zur Verfuegung. Es werden maximal 8 SIM-Baugruppen unterstuetzt.

Vorsicht: Es duerfen nicht ein E/A-Board mit 16 KB Mailbox und ein SIM-Board mit derselben Boardnummer verwendet werden.

- TAK (Terminal Anschluss Konzentrador)

Der TAK wird als externes Konzentratormodul den SINIX-Endgeraeten vorgeschaltet und bietet maximal vier SINIX-Peripherie-Schnittstellen, wahlweise SS97 oder V24. Es koennen maximal zwei TAKs an einem AFP-2-Draht-Bus angeschlossen werden, d.h. vier TAK's pro SIM.

Konfigurationsbeispiel:

Multibus I
SIM

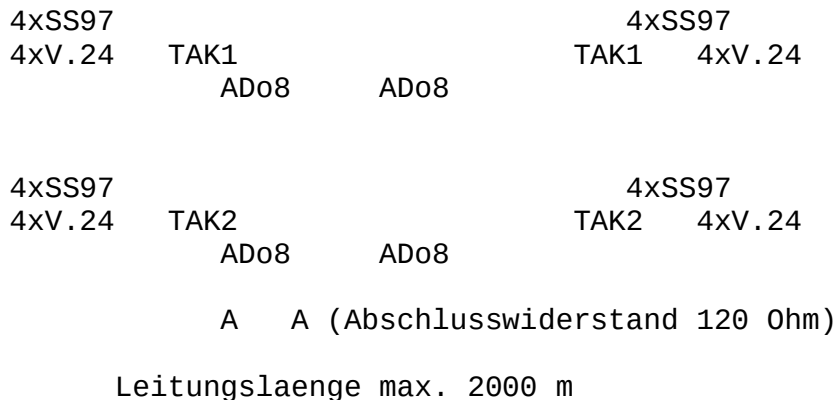
Filter

Anschlussfeld

Leitung 0

Leitung 1

Rangierverteiler



Datenuebertragungsnetz

Das Datenuebertragungsnetz basiert auf einer ungeschirmten 2-Draht-Leitung (Telefonnebenstellen-Netz) als AFP-2-Draht-Bus mit einer maximalen Laenge von 2000 Metern. Die Reichweite ist abhaengig vom Kabeltyp und der Leitungsqualitaet. Die Datenuebertragung zwischen dem SINIX-Mehrplatzsystem und den SINIX-Endgeraeten erfolgt halbduplex mit dem Alternierenden-Flanken-Puls-Verfahren (AFP) und wird mit der Uebertragungsprozedur HDLC-unbalanced gesichert. Die Uebertragungsgeschwindigkeit betraegt 187,5 Kbit/s.

Die Anzahl der gleichzeitig performant an einem MX500 betreibbaren Bildschirmarbeitsplaetze entnehmen Sie bitte dem Kapitel 2.4 dieser Freigabemitteilung.

Werden an ein SIM-Board 16 Terminals angeschlossen, so kann sich die Ausgaberate der Zeichen pro Sekunde pro E/A-Board um den Faktor 3 verlangsamen!

5.6 Hauptspeicher

Der minimale Hauptspeicherausbau des MX500-75 liegt bei 16 MB und kann ueber die Zwischenstufen 32 MB und 48 MB bis auf 64 MB hochgeruestet werden.

Der Hauptspeicherbedarf des Betriebssystems ist dem Hauptspeicherausbau angepasst. Der Speicherplatz wird beim Systemstart dynamisch angefordert. Sie koennen den Hauptspeicherbedarf des Betriebssystems (in Byte) ermitteln aus der Differenz der Groessen `real mem` minus `avail mem`, die beim Hochfahren des Systems an der Konsole ausgegeben werden.

5.7 Festplatten

SINIX V5.22 unterstuetzt auf dem MX500-75 nunmehr zwei weitere Festplattentypen, deren Aufteilung in maximal 6 Bereiche Sie bei der Installation frei bestimmen koennen. Standardmaessig gibt es fuer jeden Festplattentyp drei verschiedene Aufteilungen A, B und C, die mittels des Programms `installsys` bei der Installation geaendert werden koennen.

Es sind maximal 8 Plattenbereiche logisch ansprechbar, die sich teilweise ueberdecken (Partition a - h). Die Partition c bezeichnet immer den gesamten logischen Plattenbereich.

Fuer die Systemplatte muss die Partition a mindestens die Groesse 12 MB und die Partition b die Groesse Hauptspeicherausbau + 4 MB haben. Der restliche Teil der Platte kann frei dimensioniert werden.

Der root-Bereich muss immer auf der Partition a installiert werden, der Standard-Swap-Bereich liegt immer auf der Partition b der Platte, auf der das root-System installiert ist.

Als zusaetzliche Swap-Bereiche duerfen beliebige Partitionen verwendet werden. Alle Swap-Bereiche muessen zusammen mindestens zwischen 2 und 5 mal so gross sein wie der Hauptspeicherausbau, wobei darauf geachtet werden sollte, dass sie dabei moeglichst gleichmaessig ueber mehrere Platten verteilt sind.

Generell ist es sinnvoll, root- und usr-Bereich auf verschiedenen Platten zu installieren.

Micropolis 1558 - 380 MB

```

A           c
B   a     b           h           g
C   a     b           h           d     e           f

```

Darstellung der Plattenaufteilung
(die Darstellung ist nicht massstabsgetreu)

Logische Einteilung der Platte bei Auslieferung

MC1558 - 380 MB

Partition Kap. (KB)

```

/dev/is*a      8160
/dev/is*b     33656
/dev/is*c    312120
/dev/is*d      8160
/dev/is*e     54312
/dev/is*f     60432
/dev/is*g    122904
/dev/is*h    145856

```

* steht fuer 0 - 11, je nachdem um welche der maximal 12 moeglichen Platten es sich handelt.

Formatierte Kapazitaet:

```

pro Laufwerk   313,8 MB (davon nutzbar: 303,3 MB)
pro Zylinder   255   KB
pro Spur       17    KB
pro Sektor     512   B

```

```

Zylinder/Laufwerk           1224 (davon nutzbar: 1218)
Spuren/Zylinder             15
Sektoren/Spur               35 (davon nutzbar: 34)

```

Micropolis 1568 / Imprimis WREN VI - 760 MB

```

A           c
B    a     b           h           g
C    a     b           h           d     e           f
  
```

Darstellung der Plattenaufteilung
(die Darstellung ist nicht massstabsgetreu)

Logische Einteilung der Platte bei Auslieferung

MC1568 / - 760 MB
WREN VI

Partition	Kap. (KB)
/dev/is*a	12320
/dev/is*b	36960
/dev/is*c	648720
/dev/is*d	12320
/dev/is*e	118848
/dev/is*f	240480
/dev/is*g	371656
/dev/is*h	225376

* steht fuer 0 - 11, je nachdem um welche der maximal 12 moeglichen Platten es sich handelt.

Formatierte Kapazitaet:

pro Laufwerk	645,5	MB (davon nutzbar: 631,2 MB)
pro Zylinder	397,5	KB
pro Spur	26,5	KB
pro Sektor	512	B

Zylinder/Laufwerk	1632 (davon nutzbar: 1626)
Spuren/Zylinder	15
Sektoren/Spur	54 (davon nutzbar: 53)

Die Groessenangaben beziehen sich auf Platten ohne Dateisystem. Das Einrichten eines Dateisystems beansprucht ca. 15% der formatierten Plattenkapazitaet.

Bei der Systeminstallation werden die Festplatten erkannt und gemaess der Kundenwuensche ins System eingebunden. Beachten Sie, dass bei Einsatz von INFORMIX-Turbo die hierfuer verwendeten Partitions nicht gemountet werden duerfen.

Hinweis: Die bisher schon mit aelteren SINIX-Versionen unterstuetzten Festplatten M2333K, M2344K und MegaFile1300 werden von SINIX V5.22 weiterhin unterstuetzt. Die genauen Angaben zu den Plattenkapazitaeten und Partitiongroessen finden Sie in der Freigabemitteilung zu SINIX V5.21!

5.8 5 1/4 Zoll Diskettenlaufwerk

Es koennen 5 1/4 Zoll "double sided" "double density" Disketten mit einer Spurdichte von 96 tpi (Spuren pro Zoll) verwendet werden. Auf MX500-70/-75/-80/-85 koennen auch "high density" Disketten verwendet werden, die dann jedoch mit /etc/flformat -c H formatiert werden muessen.

5.9 Magnetbandkassettenstreamer

Der Magnetbandkassettenstreamer dient vorwiegend als Sicherungsmedium fuer die System- und Benutzerdaten. Ausserdem wird das Betriebssystem ueber eine Magnetbandkassette installiert.

Der Magnetbandkassettenstreamer wird ueber die Geraetedateien /dev/rts0 und /dev/rts8 angesprochen. Waehrend bei Angabe von /dev/rts0 beim close automatisch zurueckgespult wird, bleibt das Band bei Angabe von /dev/rts8 hinter der Tapemark stehen bzw. es wird hinter die den Daten folgende Tapemark positioniert.

In SINIX V5.22 wird vom Systemkern ein Puffer von 1000 KB zur Beschleunigung der Ein-/Ausgabeoperationen am Streamer verwendet. Bei allen Rechnern mit der MMU 32382 wird der Puffer erst beim Eroeffnen der Geraetedatei reserviert.

Es werden Magnetbandkassettenstreamer mit 9 Spuren (max. 60 MB) und 18 Spuren (max. 150 MB) unterstuetzt. In den Rechnermodellen MX500-75/-85 sind standardmaessig die 18-Spur-Laufwerke eingebaut, mit denen auch die 9-Spur-Kassetten gelesen werden koennen.

Kassetten, die auf 18-Spur-Laufwerken beschrieben wurden, sind auf anderen Laufwerken nicht mehr lesbar.

Format der Magnetbandkassette: 1/4 Zoll

Kapazitaet der Kassette: 60 MB oder 150 MB

Bestellnummer: U7-H3 (fuer beide Laufwerkstypen)

5.10 1/2 Zoll Magnetbandgeraet FS1000/FS2000

Neben den Funktionen des Magnetbandgeraetes FS1000 (3504-160) unterstuetzt das Magnetbandgeraet FS2000 (3504-625) auch Schreibdichten von 800 und 6250 bpi.

Die zum Arbeiten mit dem Magnetbandgeraet zur Verfuegung stehenden Geraetedateien sind im Systemverwalterhandbuch zu SINIX V5.22 beschrieben.

Wird eine Geschwindigkeit von 50 ips gewaehlt, so wird das Band gleichmaessiger transportiert (streaming mode) als bei einer Geschwindigkeit von 100 ips.

Tritt beim Lesen/Schreiben auf das Magnetbandgeraet ein Fehler auf, so muessen Sie unter Umstaenden 30 Minuten auf ein Timeout des Treibers warten bis das Magnetbandgeraet wieder ansprechbar ist. Die Zeitdauer bis zum Timeout muss so lang gewaehlt werden, damit ein ggf. langlaufendes Kommando nicht abgebrochen wird und die folgenden Kommandos fehlerfrei ausgefuehrt werden koennen.

5.11 Magnetbandkassettenlaufwerk 2,3 Gbyte (Video-8)

Mit SINIX V5.22 wird erstmals ein Video-8-Magnetbandkassettenlaufwerk unterstuetzt (Format: 5 1/4 Zoll).

Das Magnetbandkassetten Laufwerk dient als Sicherungsmedium fuer System- und Benutzerdaten.

Der Anschluss des Laufwerks an den MX500 erfolgt ueber den NCR Hostadapter ADP32-03.

Die Daten werden auf speziellen 8-mm Video Kassetten im sogenannten Helical-Scan-Verfahren aufgezeichnet. Die aufgezeichneten Spuren sind 0.025 mm breit, der durchschnittliche Spurabstand betraegt 0.031 mm.

Folgende Geraetedateien stehen zum Arbeiten mit dem Magnetbandkassettenlaufwerk Video-8 zur Verfuegung:

	interne Geraetedatei	Filemark Pufferung bei close	Kassetten- auswurf bei close	rewind
/dev/exa0	ja	ja	nein	ja
/dev/exa0v	nein	ja	nein	ja
/dev/exa8	ja	ja	nein	nein
/dev/exa8v	nein	ja	nein	nein

Die Geraetedatei /dev/exa ist ein symbolischer Link auf die Geraetedatei /dev/exa0.

Bei den Geraetedateien exa0 und exa8 erfolgt eine Zwischenpufferung der Daten im Treiber, wohingegen bei den Geraetedateien exa0v und exa8v keine Pufferung der Daten erfolgt. Um den Streaming-Modus des Laufwerks aufrecht zu erhalten muss eine Zwischenpufferung im Treiber erfolgen. Daher sollten nach Moeglichkeit nur die Geraetedateien /dev/exa0 und /dev/exa8 verwendet werden.

Es koennen dieselben Kommandos wie fuer den Magnetbandkassettenstreamer benutzt werden: tar, dump/restore, dd, mt, cp . . .

Beispiele:

a) /etc/dump 0fs /dev/exa0 122000 /usr1

b) tar cvf /dev/exa8 /usrX (Archiv 1)
tar cvf /dev/exa8 /usrY (Archiv 2)
tar cvf /dev/exa0 /usrZ (Archiv 3)

Nach dem letzten tar-Kommando erfolgt ein rewind. Mit Hilfe des Kommandos mt kann nun auf die einzelnen Archive positioniert werden (vgl. Kapitel 2.7.7):

c) mt -f /dev/exa8 fsf 2 (positioniert auf Archiv 3)
tar tvf /dev/exa8 (Inhalt des 3.Archivs)

d) Sicherung auf Folgebaender:
ucb tar cvfk /dev/exa0 2097100 /usrX

Mit dem Schalter k wird die Kapazitaet des Video-8-Bandes in KByte (entspricht 2048 MByte) angegeben. Somit wird ggf. ein Folgeband angefordert.

Spurformat

Jede Spur besteht aus 8 1K-Datenbloecken. Ein Formatieren des Bandes ist nicht erforderlich. Beim Schreiben auf Band wird ein 'Read after Write' - Check durchgefuehrt.

Technische Daten

Kapazitaet (brutto):	2332	Mbyte
Leistung:	246	KBytes/sec
Bandgeschwindigkeit:	10.89	mm/sec (0.429 ips)
Rotationsgeschwindigkeit:	1800	rpm

Kassettyp:	2048 MByte-Kassette
Bestellnummer	U1-H25
Bestellnummer der Reinigungskassette:	U50-H22

5.12 DUE-Baugruppen

Es koennen bis zu 6 DUE-Baugruppen mit 256 KB oder mit 1 MB Speicherausbau eingebaut werden. Ueber diese Baugruppen kann sowohl Stations- als auch Rechnerkopplung betrieben werden.

Fuer den Betrieb einer ladbaren DUE-Baugruppe wird CCP-STA1, CCP-STA3, CCP-STA4, CCP-WAN1, CCP-WAN2, CCP-WAN3, CCP-WAN4, CCP-WAN5 oder CCP-WAN6 benoetigt. CCP-STA3, CCP-WAN3 und CCP-WAN4 ist nur auf Sonderfreigabe erhaeltlich. Mit CMX V2.5A koennen Anwendungen ablaufen oder erstellt werden, die ueber das installierte CCP mit Ihrem Rechner kommunizieren. Aufsetzend auf CMX kann mit dem Softwareprodukt EMDS ab V2.1B Terminal-(9750) und Druckeremulation und mit FT-SINIX V2.2 Filetransfer betrieben werden.

5.13 Ethernet-Prozessor

Lokale Netze (ISO) werden durch die Softwareprodukte CCP-LAN2 V1.5 zusammen mit REMOS2 bzw. EMDS und FT-SINIX unterstuetzt.

6 COLLAGE

Das Laufzeitsystem COLLAGE-R V3.0B ist Bestandteil von SINIX V5.22. Es ist ablauffaehig auf den Terminals 97801 und 97808.

COLLAGE V3.0B ist gegenueber V2.0 u.a. um folgende Punkte erweitert:

neuer Applikationsmonitor, neue Program Information Files, erhoehnte Fensteranzahl, Arbeitsstandsicherung des Desktops, Mehrfachselektion, Fenster-Fenster-Kommunikation, erweiterte Gestaltung eigener Benutzeroberflaechen, COLLAGE auf Terminal 97801 (Alpha-COLLAGE), LAN-Faehigkeit von Alpha-COLLAGE.

Im Terminal 97808 steht aufgrund des ausgelagerten Applikationsmonitors fuer Anwendungen mehr Speicherplatz zur Verfuegung.

6.1 Technische Hinweise

Im Anschluss an die COLLAGE-Installation muessen Sie fuer jedes Grafik-Terminal die Terminal-Firmware neu laden.

Dabei gehen Sie folgendermassen vor:

- Einloggen unter einer Shell-Benutzerkennung im xopen- oder sie-Universum (nicht root!)
- Terminal 97808 ausschalten
- Return-Taste druecken und festhalten
- Terminal einschalten
- Return-Taste erst loslassen, wenn sich das Terminal mit "TEST END" gemeldet hat
- Aufruf: att /usr/lib/blit/ansiload
Achtung: zu diesem Zeitpunkt arbeiten Sie mit internationaler Tastaturbelegung, d.h. "/" befindet sich rechts auf der untersten Tastaturzeile!)
- nach einiger Zeit erscheint die Meldung:
"Terminal 97808 <Versionsdatum>"
- nochmals ausloggen (END-Taste druecken), sodass "getty" und "keyload" fuer dieses Terminal neu gestartet werden.

6.2 Produkt-Einsatz

- COLLAGE ist nur im xopen-Universum ablauffaehig.
- COLLAGE V3.0 ist gleichzeitig in Deutsch und in Englisch ablauffaehig, abhaengig von der Belegung der Umgebungsvariablen LANG.
Im xopen-Universum gilt standardmaessig LANG=En_US.ASCII.
Wollen Sie Deutsch als Voreinstellung, so setzen Sie LANG=De_DE.646 in /usr/att/etc/profile bzw.

in \$HOME/.profile.

- Hinweise fuer den Systemverwalter:

Es wird empfohlen, die Anschlusse der Grafik-Terminals auf moeglichst viele E/A-Boards zu verteilen.

Haben Sie einen Benutzer als COLLAGE-Benutzer mit bzw. ohne Shellberechtigung eingetragen, so sollten Sie in der Datei /etc/passwd das Startprogramm "/usr/bin/collage" bzw. "/usr/bin/collmen" aendern in "/usr/bin/collage.pw" bzw. "/usr/bin/collmen.pw". Nur dann werden beim Einloggen dieser Benutzer auch die Dateien /etc/profile und \$HOME/.profile ausgefuehrt.

Damit COLLAGE richtig geladen werden kann, muss der Terminal-Eintrag in /etc/termcap das Landeskennzeichen "LK=..." (z.B. LK=deut) enthalten.

- Alpha-COLLAGE:
Sie starten Alpha-COLLAGE mit dem Aufruf "acollage". Ist ein Anwender als COLLAGE-Benutzer eingetragen (Startprogramm /usr/bin/collage in /etc/passwd), so kann er gleichzeitig an Terminals 97808 und 97801 arbeiten. Die Prozedur "collage" entscheidet u.a. in Abhaengigkeit von der TERM-Variablen, ob die Grafik- oder Alpha-Version von COLLAGE gestartet wird.

6.3 Einschränkungen

- Waehrend COLLAGE laeuft, duerfen Sie auf Ihrem Rechner das Datum bzw. die Uhrzeit nicht verstellen. Dies sollte grundsaeztlich nur im Single-User-Modus erfolgen, um Inkonsistenzen zu vermeiden.
- Wenn Sie an mehreren 97808-Terminals mit sehr komplexen Anwendungen arbeiten, wird empfohlen, zusaetzliche Swap-Bereiche (weitere Platten) zu verwenden.
- Wenn Sie die Terminalemulation em9750 unter COLLAGE aufrufen, duerfen Sie waehrend des Verbindungsaufbaus kein anderes Fenster aktivieren.
- Hintergrundprozesse oder Daemonen duerfen Sie in einem COLLAGE-Fenster nur dann starten, wenn Sie fuer diese Prozesse die Kanaele fuer stdin, stdout und stderr schliessen oder umlenken.
Bsp.: proz >/dev/null 2>&1 &

Fuer Anwendungen, die einen Daemon starten, muss gewaehrleistet sein, dass nach dem fork-Aufruf alle von der Anwendung ererbten geoeffneten File-Deskriptoren im Kindprozess geschlossen werden. Dies laesst sich folgendermassen erreichen:

```
include <stdio.h>

fork();

if (sohn)
    /* alle Filedeskriptoren schliessen */
    for (i=0;i<_NFILE;i++)
        close(i);
        exec(daemon);
```

- Auf Grafik-COLLAGE V3.0B koennen Sie systemweit maximal 255 Fenster oeffnen (Summe von Old- und New-Style-Fenstern).
- Fuer metaprintc ist die maximale Hoehe bzw. Breite eines Metafiles 4096 Pixel bei vertikalem bzw. horizontalem Ausdruck.
- Innerhalb von COLLAGE wird der Bereitstellungsbereich G2 (fuer besondere Zeichensaetze) nicht unterstuetzt. Old-Style-Anwendungen, die auf G2 zugreifen, laufen deshalb nur mit Einschraenkungen unter COLLAGE.

Alpha-COLLAGE:

- Eine HARDCOPY loesen Sie in Alpha-COLLAGE nicht mit CTRL PRINT, sondern mit PRINT oder SHIFT PRINT aus, nachdem Sie im Menue "Anwendungen" den Punkt "Bildschirmabzug" ausgewaehlt haben.
- In Alpha-COLLAGE sollten Sie nicht die CHCODE-Taste betaetigen. Wenn Sie zwischen nationalem und internationalem Zeichensatz wechseln wollen, verwenden Sie im Menue STANDARD die Auswahl CHCODE. Siehe Manual "Bedienen Verwalten Programmieren (B-V-P)", S. 7-78.
- Funktionstasten "gesamter Bildschirm" (U22; Taste unterhalb von F22) und "zum Symbol verkleinern" (F20):

wenn Sie ein OS-Fenster auf den ganzen Bildschirm ver-groessern (U22) und danach zum Symbol verkleinern (F20), so verschwindet das ikonisierte Fenster hinter der Meneuleiste. Um dieses Fenster wieder sichtbar zu machen, muessen Sie entweder wieder die Taste "gesamter Bildschirm" druecken oder den Zeiger an die Stelle positio-

nieren, an der Sie das Fenster haben wollen, und die Taste "neue Position" (F21) betätigen. Um das Fenster wieder auf die volle Größe aufzuziehen, drücken Sie nochmals F20 "zum Symbol verkleinern".

- Emulation 9750 unter Alpha-COLLAGE:
(betrifft EMDS V3.0)
Um in Alpha-COLLAGE mit der Emulation 9750 arbeiten zu können, müssen Sie eine Prozedur mit folgendem Inhalt in APPLICATIONS eintragen bzw. in einem Shell-Fenster starten:

```
TTYNAME= tty
export TTYNAME
em9750
echo "Weiter mit RETURN"
read x
```

Eintrag des Prozedurnamens in der Datei PROGRAMS nicht vergessen! (s. Manual "B-V-P", S. 4-27)

- Wenn Sie zusätzliche Menues in die Menueleiste einhängen, so müssen deren Namen mehr als ein Zeichen enthalten.
- HIT V4.0 (Old Style): Starten Sie HIT in einem Standardfenster, so müssen Sie eine Prozedur starten, in der vor dem Hit-Aufruf die Variable HITTERM=D.C2 gesetzt wird (usr/bin/HIT/HITV4.0-D). Dies gilt nicht, wenn Sie HIT in einem "full screen window" (gesamter Bildschirm) starten. Dort rufen Sie HIT genauso auf, wie ausserhalb von COLLAGE.

Ein Wechsel zwischen Standardfenster und Fenster mit voller Bildschirmgröße (Taste U22) während einer HIT-Sitzung bringt Probleme mit dem Zeichensatz. Sie sollten sich für einen Modus entscheiden.

- Nach dem Beenden von Alpha-COLLAGE ist der ladbare Teil von G0 ueberschrieben. Sie müssen sich erneut einloggen, um wieder auf Sonderzeichen in G0 (z.B. Umlaute) zugeifen zu können.

Hinweise zum Manual "B-V-P":

- Das Beispiel zu DESKTOOLS auf den Seiten 4-14 und 4-20 funktioniert nur dann, wenn "sie" in PROGRAMS eingetragen ist.
- Namen für DESKTOOLS dürfen keinen Punkt enthalten.
- Die Anzahl der Einträge in APPLICATIONS ist auf 22 beschränkt.

- Wenn Schluesselwoerter in PIFs Unterstriche enthalten, so werden diese in Auswahlfenstern als Leerzeichen dargestellt.
- Seite 7-16: Funktionstaste "DESKTOP: Letztes Objekt" (F4). Diese Taste bringt den Zeiger zum ersten Objekt der letzten sichtbaren Zeile im aktuellen Fenster.
- Seite 7-25: Aktives Fenster zum Symbol verkleinern. Ein verkleinertes Fenster kann wieder geoeffnet werden, indem Sie den Zeiger auf sein Symbol stellen und dann die Taste START druecken, bzw. - wenn es ein OS-Fenster ist - dadurch, dass Sie erneut die Taste ZUM SYMBOL VERKLEINERN druecken.
- Seite 7-52: Seitenweise lesen. Druecken Sie RETURN, wenn Sie zeilenweise vorgehen moechten; wenn Sie seitenweise blaettern wollen, druecken Sie die LEERTASTE; mit END blaettern Sie um halbe Seiten weiter.

6.4 Verhalten im Fehlerfall

Hilfe bei der Fehlersuche:

collage -d (Nur fuer Terminal 97808!):
Rufen Sie COLLAGE mit dem Schalter "-d" auf, so werden saemtliche Aktionen des COLLAGE-Servers in der Datei "s.out" mitprotokolliert.

Beim Terminal 97808 kann es in seltenen Faellen vorkommen, dass COLLAGE in einen Zustand geraet, bei dem keinerlei Reaktion mehr auf irgendwelche Eingaben erfolgt. Dann hilft unter Umstaenden nur noch, das Terminal auszuschalten und nach ca. 1 Minute wieder einzuschalten. Der Prozess "colserv" ueberprueft regelmaessig, ob der COLLAGE-Prozess im Terminal noch laeuft und beendet sich ordnungsgemaess, falls das nicht der Fall ist. Sie duerfen daher den Prozess "colserv" nicht mit "kill" beenden!

Im Fehlerfall kann die Datei "s.out" fuer den Systemdienst hilfreich sein.

6.5 Software-Erweiterungen

Im folgenden werden nur die Erweiterungen bzw. Verbesserungen gegeneuber der Vorgaengerversion COLLAGE V2.0 beschrieben.

6.5.1 Alpha-COLLAGE

COLLAGE V3.0B laeuft mit eingeschraenkten grafischen Funktionen auf dem Terminal 97801.

Die Bedienung von Alpha-COLLAGE erfolgt durch Cursor-Tasten, RETURN-, START-, DEL-, HILFE- und ENDE-Taste bzw. durch die Funktionstasten (Tastaturmaske).

Verwendung von Alpha-COLLAGE s. Manual "B-V-P", Kap. 7.

Tasten

Achtung: Damit die Tasten MENU, END und U18 - U22 (unter F18 - F22) nicht von COLLAGE abgefangen, sondern an Old-Style-Applikationen durchgereicht werden (z.B. ced), muss im Menue "Standard" die Funktion "U-Tasten uebergeben" aktiviert werden.

Diese Aktivierung gilt waehrend der gesamten COLLAGE-Sitzung solange, bis Sie die Auswahl "U-Tasten abfangen" markieren, und wird beim Verlassen von COLLAGE gesichert. Siehe Manual "B-V-P", S. 7-79. Beachten Sie bitte, dass Sie jedesmal mit der Taste MODE in den Systemmodus umschalten muessen, wenn Sie ein COLLAGE-Menue aufziehen moechten.

Die Belegung der Funktions-Tasten fuer Alpha-COLLAGE kann systemweit oder benutzerspezifisch konfiguriert werden. Siehe dazu Manual "B-V-P", Abschnitt 8.3.

Anwendungen auf Alpha-COLLAGE

Anwendungen, die mit dem Entwicklungssystem von COLLAGE V3.0B uebersetzt und gebunden wurden, koennen sowohl auf der alphanumerischen Bildschirmeinheit 97801 als auch der grafischen 97808 ablaufen. Diese Anwendungen erscheinen mit einer einheitlichen Bedienoberflaeche und werden alternativ mit Cursor (97801) oder Maus (97808) bedient. Reine grafische, pixelorientierte Funktionen lassen sich natuerlich nicht auf die 97801 abbilden.

Das Verhalten von Alpha-COLLAGE ist im Full-Screen-Modus (Taste "Gesamter Bildschirm") performanter als im normalen Modus.

6.5.2 Betrieb ueber LAN

Ein Betrieb von Grafik-COLLAGE ist mit der aktuellen Version ueber LAN nicht moeglich.

Alpha-COLLAGE koennen Sie ueber LAN an einem fernen Rechner aufrufen.

In diesem Fall erhalten Sie ein kleines Menue mit folgenden Auswahlen:

1. Ende
2. Alpha-COLLAGE an einer Bildschirmeinheit 97801
3. Alpha-COLLAGE innerhalb COLLAGE
4. Shell

Mit Auswahl "1" kehren Sie in Ihr Ausgangsprogramm zurueck. "2" waehlen Sie, wenn Sie Alpha-COLLAGE an einem Terminal 97801 oder 97808 betreiben wollen.

Wenn Sie Alpha-COLLAGE aus einem COLLAGE-Fenster heraus aufrufen, waehlen Sie Nummer "3".

Mit der Auswahl "4" wird am fernen Rechner nicht Alpha-COLLAGE, sondern eine Shell gestartet (nuetzlich bei rlogin als COLLAGE-Benutzer).

Hinweis:

Ein rlogin als COLLAGE-Benutzer sollte ausserhalb von COLLAGE an einer Bildschirmeinheit 97801 oder 97808 mit anschliessender Auswahl "2" erfolgen.

In den anderen Faellen bestehen Einschraenkungen bei der Darstellung von nationalen Sonderzeichen und der CHCODE-Funktion.

TTYNAME:

Beim Aufruf von Alpha-COLLAGE wird die Variable TTYNAME auf folgenden Wert gesetzt:

```
TTYNAME=hostname:tty(50000)xx
```

```
hostname Rechnername im LAN-Netz
ttyxx Name des Terminals
```

```
z.B. TTYNAME=zugspitze:tty(50000)02
```

6.5.3 Der Applikationsmonitor

Der Applikationsmonitor (APM) ist in COLLAGE V3.0 als COLLAGE-Anwendung aus dem Grafik-Terminal ausgelagert. Deshalb steht gegenueber COLLAGE V2.0 mehr Speicherplatz im 97808-Terminal zur Verfuegung.

Es gibt 3 Moeglichkeiten, den zu startenden APM zu bestimmen:

- beim Aufruf von "colrun" durch die Option "-m monitorname" (nur am Terminal 97808)
- durch Setzen der Umgebungsvariablen APM (z.B. APM=sh) (Naeheres s. Abschnitt "Bedienoberflaeche")
- durch Verwenden des internen Defaults: APM

Die Prioritaeten dabei sind:

- explizite Angabe beim Aufruf vor globaler Variable
- falls weder eine Angabe beim Aufruf vorhanden ist noch die Variable APM gesetzt ist, wird der interne Default verwendet.

Zum APM siehe auch Manual "B-V-P", Abschnitt 3.2.

6.5.4 Bedienoberflaeche

In V3.0 existiert fuer Standard-Anwendungen ein eigenes Menue, das nur eingehaengt wird, wenn eine Standard-Anwendung (Old Style) im aktiven Fenster laeuft.

Eigene Erweiterungen bzw. Aenderungen der Menues sind moeglich ueber die PIFs "DESKMENU" und "APPMENU". Siehe Manual "B-V-P", Abschnitt 4.3.3.

Ausserdem koennen in COLLAGE V3.0B zusaetzliche Menues in die Menueleiste eingehaengt werden. Siehe dazu Manual "B-V-P", Abschnitt 4.5.

Wenn Sie nur die Fenstertechnik von COLLAGE nutzen wollen, koennen Sie die Shell als APM starten:

```
$ APM=sh
$ export APM
$ collage
```

Dadurch erhalten Sie eine Oberflaeche mit dem Mini-Menue "Desktop" (Start/Ende).

Nach der Auswahl von "Start" wird jeweils ein Shell-Fenster geoeffnet, mit "Ende" koennen Sie COLLAGE wieder beenden.

COLLAGE sichert den Arbeitsstand des Benutzers beim Beenden und stellt diesen Zustand beim Starten wieder her. Dabei werden die Arbeitsstandsicherungen eines Benutzers fuer Grafik- und Alpha-COLLAGE getrennt verwaltet. Ausserdem hat der Benutzer die Moeglichkeit, seine COLLAGE-Bedienoberflaeche ueber bestimmte Shell-Variable zu konfigurieren. Siehe dazu Manual "B-V-P", Kap. 4.3.

6.5.5 Rechter Mausknopf

In COLLAGE V3.0 ist es moeglich, in Verzeichnisfenstern mehrere Objekte gleichzeitig zu selektieren.

Aenderung fuer den Benutzer:

Um die Menueleiste zu erreichen, genuegt es nicht mehr, den rechten Mausknopf zu druecken. Der Zeiger muss in die Menueleiste hineingefuehrt werden.

Weitere Informationen zum Arbeiten mit der Maus finden Sie im Manual "B-V-P", Abschnitt 2.1.

6.5.6 PIF - Program Information Files

In COLLAGE V3.0 gibt es folgende Program Information Files, die z.T. neu, z.T. mit geaenderter Syntax sind:

APMMENU
 APPLICATIONS
 DESKMENU
 DESKTOOLS
 DIRTYPES
 ICONTYPES
 METHODS
 OBTYPES
 PROGMETHODS
 PROGRAMS
 PROGTYPES

Beschreibung siehe Manual "B-V-P", Abschnitt 4.3.3.

Hinweis:

Die Syntax der Datei PROGRAMS ist erweitert worden. Als x,y-Koordinaten, Hoehe und Breite fuer Old-Style-Fenster koennen Sie jeweils den Wert -1 angeben. COLLAGE waehlt dann folgende Standard-Werte fuer das Fenster, abhaengig davon, ob Grafik- oder Alpha-COLLAGE laeuft:

x, y, Hoehe, Breite
 Grafik-COLLAGE (ISAR16) 20, 40, 640, 400
 Alpha-COLLAGE 32, 16, 560, 350

Beachten Sie bitte bzgl. Alpha-COLLAGE, dass COLLAGE intern die Pixel-Koordinaten umwandelt in Alpha-Zeichen-Koordinaten und dabei rundet.

Die PIFs mit deutschen Texten stehen standardmaessig im Dateiverzeichnis /usr/lib/col/PIF/De bzw.
 /usr/lib/col/PIF/collmen/De.

Achtung: Variable LANG=De_DE.646 setzen!

Der Pfadname fuer individuelle PIFs muss ueber die Variable PIF_PATH definiert werden.

6.5.7 ICONS

Die Icons fuer Grafik-COLLAGE stehen standardmaessig im Dateiverzeichnis /usr/lib/col/ICONS, die fuer Alpha-COLLAGE in /usr/lib/col/Icons.

In beiden Dateiverzeichnissen werden auch verkleinerte Symbole (16x18 Pixel) angeboten. Mit diesen koennen Sie Dateien im Verzeichnisfenster in einer kompakten Form darstellen (Menue "Ablage", Befehl "Darstellung"). Siehe dazu Manual "B-V-P", Abschnitt 4.3.1.

Der Pfadname fuer individuelle ICONS muss ueber die Variablen ICON_PATH und AICON_PATH definiert werden. AICON_PATH wird nur innerhalb von Alpha-COLLAGE benutzt, in Grafik-COLLAGE wertet der Monitor die Variable ICON_PATH aus. Wenn AICON_PATH nicht definiert ist, wird immer ICON_PATH verwendet.

Wenn Sie im Verzeichnisfenster Dateien mit bestimmtem Suffix durch ein eigenes Icon darstellen wollen, muessen Sie in der Datei ICONTYPES folgenden Eintrag vornehmen:

```
.sfx:name.icn
```

Dabei steht ".sfx" fuer das Suffix, "name" fuer den Namen Ihres Icons.

7 Das C-Entwicklungssystem

7.1 Allgemeines

Mit SINIX V5.22 wird im att-Universum ein C-Entwicklungssystem zur Verfügung gestellt, das sich im besonderen durch folgendes auszeichnet:

a) X/OPEN Portability Guide III konforme Libraries und Headerfiles.

b) Es werden der zu CES V5.21 kompatible Sprachumfang nach Kernighan/Ritchie sowie optional auch wichtige Sprachkonstrukte des zukuenftigen ANSI-C Standards unterstuetzt.

7.1.1 Systemgrenzen

Beschreibung	Bez.*)	Wert
Anzahl der Map-Struktur Eintraege	MSGMAP	160
Maximale Groesse einer Message in Byte	MSGMAX	1024
Max. Anz. Bytes pro Messagequeue	MSGMNB	2048
Max. Anz. Messagequeueidentifizier im System	MSGMNI	50
Groesse eines Messagesegments in Byte	MSGSSZ	8
Max. Anz. von Messageheadern im System	MSGTQL	160
Max. Anz. von Messagesegmenten im System	MSGSEG	5120
Max. Anz. Message-Bytes im System		40960
Anzahl der Map-Struktur Eintraege	SEMMAP	100
Max. Anz. Semaphoridentifizier im System	SEMMNI	100
Max. Anz. Semaphore im System	SEMMNS	200
Max. Anz. "Undo"-Strukturen im System	SEMMNU	100
Max. Anz. Semaphore pro Semaphoridentifizier	SEMMSL	20
Max. Anz. Semaphoroperationen pro Systemaufruf	SEMOPM	20
Max. Anz. "Undo"-Entries pro Prozess	SEMUME	20
Maximalwert, den ein Semaphor annehmen kann	SEMVMX	32767

Eine Dokumentation der Bemuehungen und Fortschritte einer Standardisierung der Programmiersprache C und viele Hinweise und Empfehlungen, wie nicht portable C-Programmierung zu vermeiden ist, finden Sie z.B. in:

"X/OPEN Portability Guide", Prentice Hall
Englewood Cliffs, New Jersey 1988

7.1.3 Beendigung des Compilers aufgrund von Speicherengpaessen

Bei der Uebersetzung von (grossen) Programmen kann es vorkommen, dass sich eine der Compilerphasen mit einer aehnlichen Fehlermeldung wie "out of memory" oder "cannot realloc" meldet, oder der Compiler meldet "cannot execute %s", wobei %s durch den Namen der entsprechenden Phase ersetzt wird.

Dieses Problem kann darin begruetet sein, dass aufgrund momentaner starker Systemauslastung nicht mehr genuegend Platz in der Swap-Partition ist. In diesem Fall wurde dann eine Speicherplatzanforderung vom Betriebssystem zurueckgewiesen oder ein fork konnte nicht ausgefuehrt werden. Da dies ein temporaeres Problem ist, kann man seine Compilation nach einiger Zeit (evtl. nach geeigneten Massnahmen des Systemverwalters) noch einmal starten. Manchmal kann das Problem umgangen werden, indem beim Compilieren der Optimizer ausgeschaltet wird.

7.1.4 Ablauffaehige Phase kann nicht erzeugt werden

Der Grund fuer die Bindermeldung "cannot create Programmname" oder "Programmname kann nicht erzeugt werden" ist haeufig der, dass eine Phase mit dem Namen <Programmname> bereits existiert und gerade ablaeuft oder aufgrund fehlender Zugriffsrechte nicht ueberschrieben werden kann.

7.1.5 Fehler zur Laufzeit des Programms

Viele Fehler sind auf eine falsche Verwendung von Zeigern in C-Programmen zurueckzufuehren. Verwenden Sie Zeiger nur dann, wenn Sie sichergestellt haben, dass diese Zeiger auf einen sinnvollen Speicherplatz zeigen.

Groessere Programme sollten in der Entwicklungsphase eine eingebaute TRACE-Funktion besitzen, die alle Programmzustaeude, Funktionsaufrufe usw. in einer TRACE-Datei protokolliert.

Ist man mit dem Debuggerprogramm adb vertraut, so kann das Programm natuerlich auch unter der Kontrolle des adb ablaufen.

Mit der eigenstaendigen Liefereinheit DBX steht ein komfortabler Source-Code Debugger zur Verfuegung. Der DBX gehoert nicht zum CES; er muss getrennt bestellt werden.

7.1.6 Verwendung von Signalen

Signale sollten zur Synchronisation von Prozessen nicht benutzt werden. Anwendungen, die solches versuchen, zeigen haeufig ein zeitabhaengiges Verhalten, insbesondere auf einem Parallelrechner. Die zur Verfuegung stehenden Mittel (`signal(2)`, `setjmp(3C)`, `longjmp(3C)`) reichen mitunter nicht aus, um die Anwendung vor diesem zufaelligen Verhalten zu schuetzen. Ein zeitabhaengiges Verhalten tritt insbesondere dann auf, wenn ein und dasselbe Signal von mehreren Prozessen an einen anderen Prozess geschickt wird. Im Normalfall sollen Signale einem Prozess eine Endebedingung anzeigen, auf die er auf geeignete Weise reagiert, wie z.B. eine Einstellung des Terminals rueckgaengig machen und sich dann beenden.

7.1.7 Exitstatus

Der Exitstatus eines Prozesses, der sich "normal" beendet, ist undefiniert, falls nicht explizit die Funktion `exit()` aufgerufen wird bzw. die Funktion `main` nicht die Anweisung "return ausdruck;" enthaelt. Nach einer ueblichen Konvention zeigt ein Exitstatus mit dem Wert 0 an, dass sich der Prozess normal beendet hat; von 0 verschiedene Werte kennzeichnen den Fehlerfall.

7.1.8 Alignments

Unterschiedliche Prozessoren erfordern haeufig verschiedene Datenausrichtungen (Alignments). Compiler nehmen darauf Ruecksicht und generieren entsprechenden Maschinencode. Portable Programmentwicklung sollte deshalb keine Annahmen ueber die spezielle Datenausrichtung machen. So ist z.B. die manchmal anzutreffende Praxis, Strukturen als "Records" anzusehen und als Ganzes in einer Datei abzulegen

```
(write(datei,struktur,sizeof(struct struktur)))
```

nicht zu empfehlen.

Bei CES V5.22 ist es so, dass der Compiler bei Strukturen Datenausrichtung vornimmt. In diesem Zusammenhang wird folgende Eigenart des Compilers erwahnt: Strukturen werden auf 2-Byte-Grenze ausgerichtet, waehrend Variablen vom Typ int oder long im allgemeinen auf 4-Byte-Grenze ausgerichtet werden. Fuer die beiden folgenden Strukturen bedeutet dies:

```
struct A  int A_a; char A_b  ;
struct B  char B_a; int B_b  ;

sizeof(struct A) = 6    sizeof(struct B) = 8 ;
```

Die Struktur A hat am Ende ein "Loch" von einem Byte und die Struktur B hat in der Mitte ein Loch von 3 Byte. Legt man ein Feld der Struktur A an, so beginnt jedes zweite Feldelement auf einer 2-Byte-Grenze, die nicht gleichzeitig eine 4-Byte-Grenze ist. Dies hat ausser eventuellen Performance-Nachteilen (dafuer aber Speicherplatz-Vorteilen) keine Auswirkungen. Die Prozessoren der "series 32000" Familie verlangen kein Alignment von Variablen. Eine geeignete Ausrichtung von Variablen bringt aber Performance-Vorteile mit sich. Eine Variable vom Typ short sollte auf 2-Byte-Grenze und Variablen vom Typ int, float oder double auf 4-Byte-Grenze ausgerichtet sein. Dies sollte man beachten, wenn man entsprechende Variablenwerte in einem dynamisch angeforderten Speicherbereich ablegt. (Nur fuer solche Speicherbereiche kann der Programmierer eine Ausrichtung des Speicherbereichs selber ausfuehren.) Die Bibliotheksfunktion malloc liefert eine auf 4-Byte Grenze ausgerichtete Speicheradresse zurueck.

7.1.9 Locks

In SINIX 5.22 sind sogenannte advisory locks realisiert um Dateien bzw. Teilbereiche einer Datei zu sperren. Bei advisory locks ist es notwendig, dass alle sich eine Datei teilenden Prozesse selbst ueberpruefen, ob ein anderer Prozess einen Bereich fuer sich reserviert haelt, bevor der Schreib- bzw. Lesezugriff gemacht wird.

7.1.10 Uebergabe von Floatparametern

Die von der Sprache C vorgeschriebene explizite Typkonvertierung kann einige unerwartete Probleme mit sich bringen.

Beispiel:

```
float f_add(p1, p2)
float *p1, *p2 ;

return(*p1+*p2) ;

float tst_func(f1, f2)
float f1, f2;

return(f_add(&f1, &f2)) ;

main()

float x, y;

for ( x=0.0, y=0.0 ; x < 20.0 ; x += 1.0, y += 2.0 )
    printf("x=%f y=%f :: %f n", x, y, tst_func(x,y)) ;
```

Beim Aufruf von `tst_func` werden die Floatwerte von `x` und `y` als Doublewerte auf den Parameterstack gebracht. In der Funktion `f_add` werden diese Doubleparameter aber als Floatwerte, also falsch interpretiert. Im Zweifelsfall kann man dieses Problem dadurch umgehen, dass man alle Gleitkommavariablen als `double` definiert. In bestehenden Programmen kann man dies durch ein sed-script machen:

```
"sed -e 's/float/double/g' infile >outfile"
```

(Dieses sed-Script funktioniert natuerlich nur, falls der String `float` nicht noch anderweitig verwendet wird!)

Das oben beschriebene Verhalten des Compilers aufgrund der expliziten Typkonvertierung entspricht der X/OPEN Definition.

7.2 Spezielle Hinweise zur Verwendung des CES

7.2.1 Optimizer

Der Optimizer entfernt unter anderem "toten Code", berechnet Schleifeninvarianten ausserhalb der Schleife und optimiert die Verwendung der Prozessor-Register.

Dieser Optimizer sollte nur in Spezialfaellen nicht verwendet werden, wie z.B. im folgenden:

In einem "shared memory"-Segment wird Zustandsinformation (z.B. einfache Locks) gehalten. Wenn ein Prozess solch eine Zustandsvariable in einer Schleife abprueft (z.B. Warten auf Freiwerden eines Locks), so kann es vorkommen, dass der Optimizer diesen Wert am Anfang der Schleife in ein Prozessor-Register holt und in der Schleife immer nur mit dem Wert im Register gearbeitet wird. Der Prozess nimmt eine Veraenderung (durch einen anderen Prozess) im "shared memory"-Segment nicht wahr. Ein "spinlock" kann auf diese Weise zur Endlosschleife werden.

Bsp.: Die Variable lock sei vom Typ (short *) und sei mit einer Adresse in einem "shared memory"-Segment versorgt. In der folgenden "spin loop" wird solange gewartet, bis in dem Speicherbereich, auf den "lock" zeigt, eine "0" steht.

```
while (*lock)
    ;          /* do nothing */
```

Uebersetzt man solchen C-Code mit Optimizer, besteht die grosse Gefahr einer Endlosschleife.

Solche Sonderfaelle programmiert man deshalb am besten in einer kleinen Funktion in einem separaten Modul, das man dann ohne grosse Performance-Verluste ohne Optimizer uebersetzen kann.

Bei Verwendung von Float-Variablen kann es vorkommen, dass durch den Optimizer schlechterer (langsamerer) Code erzeugt wird.

7.2.2 Praeprozessor

Der Praeprozessor laesst einige (durchaus uebliche) Konstruktionen nicht mehr zu. Dadurch ist er kompatibel zum zukuenftigen ANSI-Standard. Nach den folgenden Praeprozessor-Anweisungen darf deshalb kein Text stehen:

```
else
endif
```

Konstruktionen der Art

```
ifdef  DEBUG
:
:
endif  DEBUG
```

fuehren zu der Fehlermeldung "invalid characters following command". Man kann dieses Problem bei einer groesseren Portierung durch das Compilerflag "-tp" umgehen. Diese Option kann in zukuenftigen Versionen nicht garantiert werden. Da die oben beschriebenen Konstruktionen im zukuenftigen ANSI-Standard nicht zulaessig sind, sollte man langfristig den Code entsprechend umstellen. (Etwa den unzuessaessigen Text als Kommentar markieren.)

7.2.3 Temporaere Compilerdateien

Der Compiler legt seine temporaeren Dateien im Dateiverzeichnis /tmp ab. Allen Teilen des Compilers, bis auf den Optimizer und den Codegenerator, kann ein anderes Dateiverzeichnis fuer temporaere Dateien zugewiesen werden. Dies geschieht ueber die Umgebungsvariable "TMPDIR" oder die Option "-Tpathname".

7.2.4 Leseschutz des leeren Adressraumanfangs

Die vom Binder erzeugten ablauffaehigen Programme haben am Anfang des Adressraumes einen 4 KB grossen Bereich, der vor Schreibzugriffen geschuetzt ist. Jeder schreibende Zugriff auf eine Adresse von 0 bis 4095 (z.B. ueber einen unversorgten Pointer), fuehrt deshalb zu einem Adressierungsfehler. ("Bus error", Signal SIGBUS.)

Mit der Binderoption "-MU" kann der Bereich von Adresse 0 bis 4095 auch vor Lesezugriffen geschuetzt werden. Bei Aufruf des Binders ueber das cc-Kommando gibt man diese Option als "-LMU" an. Diese Option wird nicht offiziell freigegeben; eine Gewaehrleistung wird nicht uebernommen.

7.2.5 Shared memory

Beim Erzeugen von "shared memory"-Segmenten werden im Dateiverzeichnis /usr/att/usr/tmp/SysVshmem Dateien angelegt, die in den Adressraum des Anwendungsprozesses abgebildet werden. Man sollte deshalb dafuer sorgen, dass in dem betroffenen Dateisystem immer genuegend freier Speicherbereich zur Verfuegung steht.

Das Dateiverzeichnis "SysVshmem" wird beim Systemstart von SINIX V5.22 durch die Prozedur /etc/rc.sys5 geleert.

shmget() erlaubt "shared memory"-Segmentgroessen von 0 bis 4 GB. Die Pruefung auf tatsaechlich verfuegbaren Speicher erfolgt erst bei shmat(). Eine Obergrenze fuer den verfuegbaren Speicher ist durch den virtuellen Adressraum des Prozessors gegeben (16 MB fuer CPU NSC32016 und NSC32032 mit

MMU NSC32082 und 2 GB fuer CPU NSC32332 mit MMU NSC32382 oder fuer CPU NSC32532). Eine andere Obergrenze ist durch den freien Platz in dem Dateisystem gegeben, in dem sich das Dateiverzeichnis /usr/att/usr/tmp/SysVshmem befindet. Die Groesse eines "shared memory"-Segmentes wird immer auf ein Vielfaches der logischen Pagesize (4KB) aufgerundet. shmat() rundet eine angegebene Anfangsadresse immer auf ein Vielfaches der logischen Pagesize auf.

Mit dem Kommando ipcrm koennen "shared memory"-Segmente geloescht werden.

Bsp.: ipcrm -m shm_{id} oder ipcrm -M shm_{key}

shm_{id} ist ein "shared memory identifier" und shm_{key} ist ein "key" (dezimal) zu einem "shared memory identifier". Das Kommando ipcs unterstuetzt "shared memory" nicht.

Die mit der Funktion shmget() erzeugten Dateien im Verzeichnis /usr/att/usr/tmp/SysVshmem dienen dem System im wesentlichen als Paging-Bereich fuer "shared memory"-Segmente. Deshalb vergroessert sich auch der Paging-Bereich nach dem ersten "attach" eines solchen Segmentes (Kommando /etc/pstat -T). Die Besonderheit der SINIX Implementierung von "shared memory" bringt es mit sich, dass jeder read bzw. write Systemcall auf eine "shared memory"-Datei, die in den Adressraum eines Prozesses abgebildet ist, den entsprechenden Hauptspeicherinhalt liest bzw. beschreibt. Da die von shmget angelegten Dateien am Anfang noch interne Verwaltungsinformation enthalten, sind diese Dateien immer etwas groesser als der angeforderte Speicherbereich.

Bei Benutzung von "shared memory" sollte man auch folgende Punkte beachten:

- Beim ersten Aufruf von shmat fuer einen "shared memory identifier" wird an die von shmget erzeugte Datei ein mit Nullbytes gefuellter Bereich angehaengt, der der Groesse des Segmentes entspricht. Ein haeufiges Erzeugen und Zerstoeren von "shared memory" Segmenten ist deshalb aus Performancegruenden nicht zu empfehlen.
- Bei jedem shmat werden vom Betriebssystem neue "pageable entries" aufgebaut. Deshalb sollte man ein Segment in einem Prozess nicht unnoetig mit shmat in den Adressraum abbilden und mit shmdt abhaengen.
- Bei jedem shmat wird ein Filedeskriptor belegt, der erst bei dem zugehoerigen shmdt wieder freigegeben wird.
- Bei jedem shmat mit "shmaddr=0" wird der "break" des Prozesses erhoehrt. Nach einem shmdt entsteht ein "Loch" im Adressraum dieses Prozesses. Dieses "Loch" kann nur durch einen Aufruf von shmat mit Angabe der Anfangsadresse dieses "Lochs" beseitigt werden. Ein haeufiger Aufruf von shmat mit "shmaddr=0" kann also dazu fuehren,

dass der Adressraum des Prozesses ueberlaufen wuerde und deshalb ein weiterer Aufruf von `shmat` nicht erfolgreich ist. (Ein Herabsetzen des `breaks` mit `brk(2)` ist nicht zu empfehlen, da andere Bibliotheksfunktionen ueber `malloc` oder mit `brk` dynamisch Speicherplatz allokatieren koennen. Das Herabsetzen des `breaks` kann dann in anderen Programmteilen unerwartete Folgen haben.)

- Ein "shared memory"-Segment kann nicht resident im Hauptspeicher gehalten werden. Ein entsprechender Aufruf von `shmctl` mit den flags `SHM_LOCK` bzw. `SHM_UNLOCK` ist wirkungslos. Ein "shared memory"-Segment unterliegt dem Paging. Deshalb fuehren unnoetig grosse "shared memory"-Segmente zu einer hohen Zahl von "page faults". Dieser Effekt kann zu erheblichen Performance-Nachteilen fuehren. Grosse "shared memory" Segmente verlangen eventuell eine grosse Anzahl im Hauptspeicher geladener Seiten ("working set"). Da diese durch die maximale "resident set size" begrenzt wird, kann es eventuell von Vorteil sein, die maximale "resident set size" mit `vmtune` zu vergroessern. (Siehe `vmtune` im Systemverwalter Manual.)
- Die bei `shmget` angelegten Dateien haben immer die Zugriffsrechte `0666`. Die beim Funktionsaufruf angegebenen Zugriffsrechte fuer das "shared memory"-Segment werden nicht uebernommen. Diese Zugriffsrechte werden nur von den "shared memory" Funktionen ausgewertet. Damit ist allerdings der Datenschutz prinzipiell nicht gewaehrleistet. Um dies zu aendern kann die Anwendung die Zugriffsrechte der Dateien explizit mit `chmod(2)` setzen. Dies kann allerdings ein unerwartetes Verhalten der "shared memory" Funktionen nach sich ziehen, da diese Funktionen erwarten, dass sie die entsprechende Datei zum Lesen und Schreiben eroeffnen koennen. Die "shared memory" Dateien haben die Namen `id_%x`, wobei `%x` durch die sedezimale Darstellung des "shared memory identifiers" zu ersetzen ist. Falls fuer den "key" nicht `IPC_PRIVATE` angegeben wurde, ist diese Datei auf die Datei `key_%x` gelinkt. Dabei ist `%x` durch die sedezimale Darstellung des "key's" zu ersetzen.
- Nach dem Aufruf von `shmat` fuer ein "shared memory"-Segment kann man die zugehoerige Datei `id_%x` und eventuell `key_%x` loeschen. Danach kann aber kein Prozess dieses "shared memory"-Segment seinem Adressraum neu angliedern. Ein Aufruf von `shmget` mit dem gleichen Key wird ein neues, voellig unabhaengiges "shared memory"-Segment erzeugen, sofern das flag `IPC_CREAT` gesetzt ist. Die Prozesse, die das zu der geloeschten Datei gehoernde "shared memory"-Segment ihrem Adressraum vor dem Loeschen bereits angegliedert hatten, koennen mit diesem Segment normal weiterarbeiten. Nur Aufrufe von `shmctl` oder `shmat` mit dem zugehoerigen "shared memory identifier" liefern einen Fehler oder falsche Ergebnisse, falls ein neues Segment mit dem gleichen Identifier erzeugt wurde.

- In der zu einem "shared memory" Segment gehoerenden Struktur shmid_ds wird das Element shm_nattch nicht versorgt.

Unabhaengig von diesen implementationsspezifischen Punkten muss man folgendes beachten:

In einem "shared memory" koennen mehrere Prozesse gleichzeitig Veraenderungen vornehmen. Je nach Verwendung des "shared memory" muessen die Prozesse eventuell ausschliessen, dass es zu gleichzeitigen Veraenderungen im "shared memory" kommt. Da unteilbare (atomare) Zugriffe nicht moeglich sind, muessen sich die Prozesse auf geeignete Weise synchronisieren. Dies gilt in erhoehtem Masse fuer einen Parallelrechner wie den MX500. Das Standardbetriebsmittel zur Prozesssynchronisation sind die Semaphore.

7.2.6 Semaphore

Eine intensive Verwendung der "System V"-Semaphore bringt eine hohe Systembelastung und Performancenachteile mit sich. Fuer Portierungen auf andere SINIX-Maschinen kann folgendes wichtig werden: ein parallelisierter Algorithmus, der sehr haeufige Zugriffe auf "System V"-Semaphore benoetigt, ist eventuell umso langsamer, je mehr Prozessoren er nutzt. Ausserdem kann die Systembelastung sehr hoch ansteigen. Bei der Verwendung von Semaphoren mit "Undo"-Strukturen ist der maximale "semaphore-adjust" Wert eines Prozesses zu beachten.

7.2.7 Implementierung von Systemaufrufen

Bestimmte Funktionen muessen nicht unbedingt als Systemaufrufe implementiert sein. Sie koennen auch durch Bibliotheksfunktionen auf andere vorhandene Systemdienste abgebildet werden (z.B. shmat(), shmget()). Wird ein Programm mit einem Debugger auf Fehler untersucht, so ist die Bibliothekssimulation von Systemaufrufen sichtbar.

7.2.8 Binder-Option "-u"

Die Binderoption "-u" wirkt sich zur Zeit nur auf Textsymbole, d.h. Namen von Funktionen aus. Bei Variablen hat sie keine Wirkung. Dem Symbolnamen, den man mit der Option "-u" uebergibt, muss immer ein "_" vorangestellt werden.

7.2.9 `curses()`: Alternativer Zeichensatz

Ueber das Attribut "A_ALTCHARSET" kann der alternative Zeichensatz eingeschaltet werden. Dieses Attribut ist nicht dokumentiert und gehoert nicht zum Leistungsumfang von X-OPEN. Deshalb kann bei seiner Verwendung keine Gewaehrleistung uebernommen werden.

7.2.10 Debug-Version des libc-Moduls `malloc.o` (`dbg_malloc.o`)

In SINIX V5.22 wird unter dem Pfad `/usr/lib` eine Debug-Version des libc-Moduls `malloc.o` namens `dbg_malloc.o` bereitgestellt.

Dieses Modul soll dem Benutzer helfen, Ueberschreibungsfehler von dynamisch allokierten Speicherbereichen zu suchen. Das Modul muss zu diesem Zweck zum eigentlichen Programm explizit dazugebunden werden.

Sowohl die Funktion `malloc()` als auch die Funktion `free()` wurde um Plausibilitaetspruefungen erweitert, die dafuer sorgen, dass bei einem entdeckten Fehler eine entsprechende Fehlermeldung an den Benutzer erfolgt. Anschliessend wird das Programm durch ein entsprechendes `assert()`-Statement abgebrochen. Im Einzelnen liefert `malloc()` folgende Fehlermeldungen:

- Ist die Anfangsadresse eines Bereiches, dessen Ende ueberschrieben wurde, reproduzierbar, so liefert die Fehlermeldung diese Anfangsadresse und die Groesse des allokierten Bereiches.
- Ist die Anfangsadresse eines Bereiches, dessen Ende ueberschrieben wurde, nicht reproduzierbar, so wird die ueberschriebene Adresse geliefert.

Um die gelieferten Fehlermeldungen besser auswerten zu koennen, wird bei jedem Aufruf von `malloc()` die zurueckgegebene Anfangsadresse des allokierten Bereiches sowie die Groesse des Bereiches in Form einer Meldung an den Benutzer ausgegeben, falls der Benutzer einer globalen Variablen `_silent` den Wert 0 zugewiesen hat. Defaultmaessig ist `_silent` mit 1 initialisiert. Nach demselben Schema gibt die Funktion `free()` ihren Aufrufparameter aus.

Ausserdem liefert die Funktion `free()` noch Fehlermeldungen fuer den Fall, dass

- versucht wird, einen Bereich freizugeben, der nie allokiert wurde (z.B. ungueltige Adresse als Parameter)
- versucht wird, einen Bereich freizugeben, der im Moment nicht allokiert ist (z.B. bereits freigegeben)

Da die Funktionen `calloc()` und `realloc()` ihrerseits `malloc()` aufrufen, ist auch bei Verwendung dieser Funktionen die beschriebene Unterstuetzung gewaehrleistet.

Die Moeglichkeit weitere Fehlerfaelle zu entdecken haengt von den Werten zweier globaler Integer-Variablen (`_free_check` und `_no_free`) ab, die der Benutzer entsprechend zuweisen kann. Dabei wird folgendes bewirkt:

- `_free_check == 1`:
Vom Benutzer angeforderte bzw. freigegebene Speicherbereiche werden mit `0` initialisiert, wodurch bestimmte Benutzerfehler (schreibende Nullpointer-Dereferenzierung) zu einem Programmabbruch fuehren.
- `_free_check > 1`:
Vom Benutzer angeforderte bzw. freigegebene Speicherbereiche werden mit `0` initialisiert. Weiterhin wird bei jedem `malloc()`-Aufruf ueberprueft, ob freie Speicherbereiche mit `0` initialisiert sind. Dadurch werden Ueberschreibungsfehler in Bereichen entdeckt, die noch nie allokiert oder andererseits freigegeben aber von `malloc()` noch nicht wieder vergeben wurden.
- `_free_check == 0`:
Weder die Initialisierung mit `0` noch die o.a. Ueberpruefung wird durchgefuehrt.
- `_no_free != 0`:
Vom Benutzer freigegebene Bereiche werden von `malloc()` nicht wieder anderweitig vergeben. Zusammen mit der Belegung `_free_check > 1` koennen damit auch Ueberschreibungsfehler in Bereichen festgestellt werden, die `malloc()` sonst zum Zeitpunkt des Fehlers schon wieder anderweitig vergeben haette.

Defaultmaessig ist die Variable `_free_check` mit `2` und die Variable `_no_free` mit `0` initialisiert.

7.2.11 Funktionsprototypen

CES V5.22 unterstuetzt mit dem Compilerflag `-kansi` auch ANSI-kompatible Sprachkonstrukte, u.a. auch Funktionsprototypen. Allerdings existieren fuer die Standardbibliotheksfunktionen in den entsprechenden Includefiles z.Zt. keine Funktionsprototypen. Das fuehrt dazu, dass beim manuellen Erhoehen des Warninglevels zahlreiche "Compilerwarnings" wegen fehlender Funktionsprototypen fuer die Standardfunktionen erscheinen.

7.3 Aenderungen im CES

7.3.1 Systemcalls

kill()

Die bei kill() auftretende Inkompatibilitaet tritt nur in ganz besonderen Faellen auf, in denen Programme mit gesetztem Set-UID-Bit aufgerufen und anschliessend Signale an diese Prozesse geschickt werden.

chown()

Aus Sicherheitsgruenden kann nur ein Prozess mit der effektiven UID des Superusers erfolgreich einen chown()-Aufruf durchfuehren.

exec()

Bei Aufruf dieser Funktionen mit ungueltigen Adressen fuer Programm-Pfadnamen wird dem Prozess das Signal SIGSEGV zugestellt (frueher: Fehler mit errno=EFAULT).

read(), write()

Beim Lesen/Schreiben aus/in eine(r) vollen/leeren Pipe mit gesetztem O_NONBLOCK Flag wird der Systemcall mit Fehler beendet, errno=EAGAIN (frueher kein Fehler, Returnwert 0).

uname()

Als RELEASE wird jetzt die aktuelle Release-Kennung zurueckgegeben (z.B. V5.22), als VERSION die aktuelle Version (z.B. N4). Frueher: VERSION = Release-Kennung (z.B. V5.21) RELEASE = Datum der Freigabe.

cuserid()

Es wird jetzt der Name geliefert, der mit der effektiven User-ID des Prozesses in Verbindung steht (Frueher: Benutzererkennung des Eigentuemers des aktuellen Prozesses).

7.3.2 Datentypen

Im CES V5.22 (XPG3) haben sich gegenueber der Vorgaengerver- sion (XPG2) bei verschiedenen Funktionen die Datentypen geaendert. Da die Funktionen in verschiedenen Headerfiles fuer ein Anwenderprogramm deklariert werden, kann es bei der Uebersetzung von Programmen zu Konflikten kommen, wenn im Programm Funktionsdeklarationen nach XPG2 vorhanden sind.

7.3.3 Headerfiles

Folgende Headerfiles sind in CES V5.22 neu hinzugekommen:

```
stdlib.h
termios.h (ersetzt das bisherige termio.h)
ulimit.h
utime.h
```

7.3.4 YACC

Im Rahmen der Internationalisierung wurde der yacc 8-Bit-clean gemacht und bezueglich der Verarbeitung von 8-Bit-Zeichensaetzen erweitert:

Es werden Zeichen, deren achttes Bit gesetzt ist, in Zeichenkonstanten, Kommentaren und in Zeichenketten zugelassen. Darueberhinaus koennen 8-Bit-Zeichen auch in Namen verwendet werden, die zur Beschreibung der "YACC"-Grammatik dienen. Grundsaeztlich koennen alle Buchstaben sogenannte "Terminalsymbole" sein. Welche Zeichen als Buchstaben oder nicht als Buchstaben gelten, wird zur Laufzeit von "YACC" durch Auswerten der entsprechenden "Ctype"-Tabelle ermittelt. Dabei wird auf die zu dem Zeitpunkt vorhandene internationale Datenbasis zurueckgegriffen. Dies kann unbeabsichtigte Seiteneffekte haben. Sind naemlich Zeichen (z.B. ' ' oder ' ') als Buchstaben definiert, wie dies in der Datenbasis De_DE.646 der Fall ist, in der ' ' dem 'ae' und ' ' dem 'oe' entsprechen, so sind "YACC"-Grammatiken nicht mehr auswertbar.

Dieser Effekt tritt z.B. dann auf, wenn die Umgebungsvariable LANG auf De_DE.646 gesetzt ist, wenn der YACC aufgerufen wird. Umgehen laesst sich dieses Verhalten, wenn die Umgebungsvariable LANG gar nicht oder auf En_US.ASCII gesetzt wird.