

|                            |   |            |
|----------------------------|---|------------|
| ANWENDER-<br>DOKUMENTATION | Anleitung für den<br>Systemverantwortlichen<br>Rechnerkopplung UUCP<br>für MUTOS 1700 | MOS        |
| 11/87                      |   | MUTOS 1700 |

Programmtechnische  
Beschreibung Teil 4

Anleitung für den  
Systemverantwortlichen

Rechnerkopplung UUCP  
für MUTOS 1700

AC A 7100/7150

VEB Robotron-Projekt Dresden

Ausgabe: 11/87

Die Ausarbeitung dieser Dokumentation erfolgte durch ein Kollektiv des VEB Robotron-Elektronik Dresden Stammbetrieb des VEB Kombinat Robotron.

Nachdruck und jegliche Vervielfältigung, auch auszugsweise, sind nur mit Genehmigung des Herausgebers zulässig.

Im Interesse einer ständigen Weiterentwicklung werden alle Leser gebeten, Hinweise zur Verbesserung der Dokumentation dem Herausgeber mitzuteilen.

Herausgeber:

VEB Robotron-Projekt Dresden  
Leningrader Str. 9  
Dresden 8010

(C) VEB Kombinat Robotron

Kurzreferat

Die vorliegende Schrift wendet sich an die Fachkräfte der Datenverarbeitung, die Projekte der Rechnerkopplung vorbereiten und bearbeiten, in denen der Arbeitsplatzcomputer AC A 7100/A 7150 mit dem Betriebssystem MUTOS und dem Programmkomplex zur Rechnerkopplung UUCP eingesetzt werden. Struktur, Arbeitsweise und die Implementierung des Programmkomplexes sind ausführlich beschrieben, außerdem sind die Anforderungen an die Gerätetechnik und das Betriebssystem zusammengestellt.

| Inhaltsverzeichnis  | Seite |
|---|-------|
| 1. EINFÜHRUNG   |       |
| 2. STRUKTUR UND ARBEITSWEISE DES PROGRAMMPAKETS                       |       |
| 2.1. BESTANDTEILE   | 2-1   |
| 2.2. GRUNDSTRUKTUR  | 2-2   |
| 2.2.1. Directories  | 2-2   |
| 2.2.2. Permanente Files   | 2-3   |
| 2.2.3. Temporäre Files  | 2-7   |
| 2.3. ARBEITSWEISE   | 2-12  |
| 2.3.1. Allgemeines  | 2-12  |
| 2.3.2. Vorbereitung der Übertragung                                   | 2-12  |
| 2.3.3. Verbindungsaufbau  | 2-13  |
| 2.3.4. Datenaustausch   | 2-15  |
| 2.3.5. Verbindungsabbruch   | 2-16  |
| 2.4. ENTFERNT KOMMANDO AUSFÜHRUNG                                     | 2-17  |
| 2.4.1. Vorbereitung   | 2-17  |
| 2.4.2. Ausführung   | 2-17  |
| 3. ANFORDERUNGEN AN GERÄTETECHNIK UND BETRIEBSSYSTEM                  |       |
| 3.1. ANFORDERUNGEN AN DIE GERÄTETECHNIK                               | 3-1   |
| 3.2. ANFORDERUNGEN AN DAS BETRIEBSSYSTEM                              | 3-1   |
| 4. IMPLEMENTIERUNG  |       |
| 4.1. EINBRINGEN DER NOTWENDIGEN ZUSÄTZE IN DIE BETRIEBSSYSTEMUMGEBUNG | 4-1   |
| 4.2. BENÖTIGTE DIRECTORIES UND FILES                                  | 4-2   |
| 4.3. GENERIERUNG  | 4-2   |
| 4.3.1. Einbringen der nutzerspezifischen Parameter in den Quellcode   | 4-2   |
| 4.3.2. Compilerlauf   | 4-4   |
| 4.3.3. Bildung des Zielsystems  | 4-4   |

5. ADMINISTRATIVE ARBEIT WÄHREND DER NUTZUNG

ANLAGE A ABLÄUFE WÄHREND DES VERBINDUNGSaufBAUS

ANLAGE B ABLÄUFE WÄHREND DES DATENAUSTAUSCHES

Abkürzungsverzeichnis

MUTOS - Multiuser-Timesharing-Operating-System

IFSS - Interface sternförmig seriell

AIS - Anschlußsteuerung seriell/parallel

## 1. Einführung

Das Programmpaket UUCP unterstützt den Austausch von Files zwischen 2 oder mehreren AC A 7100/A 7150 sowie die Ausführung von MUTOS-Kommandos auf einem entfernten Rechner. Der Datenaustausch erfolgt über IFSS- Kanäle der ASP.

Ein spezielles Protokoll der Paketkommunikation sichert die fehlerfreie und transparente Übertragung aller Daten. Der Prozeß der Datenübertragung läuft im Hintergrund ab.

Die vorliegende Schrift enthält eine allgemeine Beschreibung der Arbeitsweise von UUCP, der benötigten Files und Directories sowie der für die Implementierung des Programmpaketes notwendigen Abläufe.

Für das Verständnis dieses Materials ist eine Grundkenntnis des Betriebssystems MUTOS 1700 notwendig. Ferner wird die Kenntnis der Anwendungsbeschreibung für MUTOS 1700 (Punkt 6: Rechnerkopplung) sowie der entsprechenden Teile der Programmtechnischen Beschreibung, Teil 1 von MUTOS 1700 vorausgesetzt.

## 2. Struktur und Arbeitsweise des Programmpakets

~~~~~

### 2.1. Bestandteile

~~~~~

Das Programmpaket zur Unterstützung der Rechnerkopplung UUCP besteht aus vier primären und vier sekundären Programmen. Als primäre werden hier diejenigen Programme bezeichnet, die direkt am Datenaustausch bzw. der entfernten Kommandoausführung beteiligt sind. Die übrigen sekundären Programme werden benutzt, um für den Nutzer einen gewissen Service bereitzustellen.

#### Primäre Programme:

- UUCP - Erstellt ausgehend von einer Analyse des gegebenen Kommandos Arbeits-Files (Daten-Files und Kommando-Files) im Spool-Directory (/usr/spool/uucp), die die geforderte Datentransferoperation genauer charakterisieren und später von UUCICO zur Ausführung der eigentlichen Übertragung benutzt werden.
- UUX - Analysiert eine Kommandozeile zur entfernten Ausführung von MUTOS-Kommandos und erstellt im Spool-Directory entsprechende Arbeits- und Ausführungs-Files.
- UUCICO - Führt die im Spool-Directory befindlichen Arbeits-Files aus und realisiert die eigentliche Übertragung.
- UUXQT - Führt die von UUX erstellten Ausführungs-Files zur entfernten Ausführung von MUTOS-Kommandos aus.

Als Kommandointerface für den Nutzer sind nur UUCP und UUX sinnvoll. UUCICO und UUXQT werden von UUCP bzw. UUX automatisch gestartet. Für Testzwecke ist jedoch auch ein getrennter Start von UUCICO und UUXQT möglich.

#### Sekundäre Programme:

- UUNAME - Zeigt die Namen aller im lokalen System bekannten entfernten Rechner an.
- UULOG - Vereinigt mehrere LOGFILES in ein LOGFILE und zeigt die Ausführung von UUCP-Kommandos betreffenden Statusinformationen (Eintragungen im LOGFILE) an.

UUCLEAN- Löscht alte Files im Spool-Directory.

UUKILL - Führt für UUCICO Hilfsoperationen in den Phasen des Verbindungsaufbaus und -abbruchs durch (nicht für den Bediener vorgesehen).

## 2.2. Grundstruktur

~~~~~

### 2.2.1. Directories

~~~~~

Alle nachfolgend genannten Pfadnamen entsprechen den im File uucp.h im Ausgangszustand enthaltenen Standardnamen.

- /usr/src/cmd/uucp - Dieses Directory enthält alle zum Programmpaket UUCP gehörenden Quellprogramme. Die Generierung bzw. spätere Modifikation von UUCP erfolgt in diesem Directory. Dieses Directory ist für die Arbeit von UUCP nicht notwendig.
  
- /usr/lib/uucp - Dieses Directory enthält die zum Programmpaket gehörenden 8 ausführbaren Programme (siehe Punkt 2.1.) sowie bestimmte für die Arbeit benötigte Files mit Systemdaten (USERFILE, L.sys, L-devices, SQFILE, SEQF).
  
- /usr/lib/uucp/.XQTDIR - Dieses Directory wird für die entfernte Ausführung von MUTOS-Kommandos benutzt (Ausführung von Ausführungsfiles).
  
- /usr/spool/uucp - Spool-Directory; in diesem Directory errichten die einzelnen Programmkomponenten (primäre Programme) ihre für die Arbeit benötigten Files, d.h. Daten-Files (D.\*), Kommando-Files (C.\*), Ausführungs-Files (X.\*) sowie verschiedene temporäre Files. Die Files mit Informationen über den Status der ausgeführten Operationen (LOGFILE, SYSLOG) befinden sich ebenfalls in diesem Directory.
  
- /usr/spool/uucppublic - Dieses Directory wird als Login- Directory für den UUCP- Password-Eintrag benutzt. Falls es beim Empfang eines Files von einem entfernten Rechner nicht möglich ist, das File an den gewünschten Zielort zu kopieren, wird das File in das Directory /usr/spool/uucppublic/user kopiert, wobei user der Name des Nutzers ist, der die Übertragung angefordert hat.

### 2.2.2. Permanente Files

~~~~~  
Diese Files, die bestimmte Systemdaten enthalten, befinden sich in /usr/lib/uucp. Sie müssen bei der Implementierung des Programmpaketes UUCP vom Nutzer errichtet werden.

#### USERFILE

- Dieses File enthält Informationen, die die Zugriffsrechte der einzelnen dem UUCP-System bekannten Nutzer und entfernten Rechner charakterisieren. Diese Informationen, die sich in 4 Klassen teilen lassen, beinhalten:
  - (1) zu welchen Files ein Nutzer des lokalen Systems Zugriff hat,
  - (2) zu welchen Files jeder konkrete entfernte Rechner Zugriff hat,
  - (3) welchen Loginnamen zum Anmelden am lokalen System jeder entfernte Rechner benutzt,
  - (4) ob der konkrete entfernte Rechner vom lokalen System "rückgerufen" wird (call-back Option), um seine Identität nachzuweisen.

Das USERFILE enthält für jedes entfernte System (Nutzer) eine Zeile.

Das Format einer solchen Zeile ist folgendes:

```
Login,sys [c] Pfadname [Pfadname]...
```

Bedeutung der einzelnen Felder:

- Login - Login-Name für den Nutzer oder den entfernten Rechner
- sys - Systemname für den entfernten Rechner
- c - wahlweise angebbare Option. Falls diese Option angegeben wurde, wird für diesen entfernten Rechner die call-back Prozedur ausgeführt, d.h. der lokale Rechner setzt einen Ruf an den entfernten Rechner auf, um die Identität dieses Rechners zu prüfen. Die laufende Verbindung wird dabei abgebrochen und beim Rückrufen neu hergestellt.
- Pfadname - Präfix des Pfadnamens, zu dem der Nutzer (Feld Nutzer) Zugriffserlaubnis hat.

Realisierung der Kontrolle der Zugriffsrechte nach den oben genannten vier Punkten:

- (1) Wenn das Programmpaket UUCP ein Kommando ausführt, das auf dem lokalen System gegeben wurde, d.h. UUCP arbeitet im Master-Modus, dann werden als erlaubte Pfadnamen die angenommen, die in der ersten Zeile im USERFILE angegeben sind, deren Login-Name mit dem Loginnamen des Nutzers übereinstimmt, der das Kommando gab. Falls eine solche Zeile nicht gefunden wurde, wird die erste Zeile benutzt, deren Feld für den Login-Namen leer ist.
- (2) Wenn das Programm auf ein Kommando reagiert, das von einem entfernten Rechner gegeben wurde, d.h. UUCP arbeitet im Slave-Modus, dann werden als erlaubte Pfadnamen die angenommen, die in der ersten Zeile angegeben sind, deren Systemnamen mit dem Systemnamen des entfernten Rechners übereinstimmt. Falls eine solche Zeile nicht gefunden wurde, wird die erste Zeile benutzt, deren Feld für den Systemnamen leer ist.

- (3) Wenn sich ein entfernter Rechner am lokalen System anmeldet (login), muß der dabei benutzte Login-Name auch im USERFILE enthalten sein. Es können mehrere Zeilen den gleichen Login- Namen enthalten oder in einer dieser Zeilen muß der Systemname mit dem Namen des konkreten entfernten Systems übereinstimmen oder das Feld für den Systemnamen muß leer sein.
- (4) Wenn die Zeile im USERFILE, die mit (3) ausgesucht wurde, die c-Option enthält, erfolgt ein Rückruf an den entfernten Rechner, bevor irgendwelche Operationen ausgeführt werden.

#### Beispiele:

##### - Die Zeile

```
n,r /usr/dat
```

erlaubt dem Rechner r , sich mit dem Namen n am lokalen System anzumelden und einen Transfer von Files anzufordern, deren Pfadname mit "/usr/dat" beginnt.

##### - Zeile

```
otto, /usr/otto
```

erlaubt dem Nutzer otto , ein Kommando betreffs Files, deren Pfadname mit "/usr/otto" beginnt, zu geben.

##### - Die Zeilen

```
e41,labor1 /usr/dat /usr/spool
e41,      /usr/spool
```

erlauben allen entfernten Rechnern, sich mit dem Namen e41 am lokalen System anzumelden. Wenn der Name des entfernten Rechners jedoch nicht labor1 ist, kann dieser Rechner nur Transfers von Files, deren Pfadname mit "/usr/spool" beginnt, anfordern.

##### - Die Zeilen

```
root, /
, /usr
```

erlauben allen Nutzern den Zugriff zu Files, deren Pfadname mit "/usr" beginnt. Der Nutzer mit dem Login-Namen root hat die Zugriffserlaubnis zu allen Files.

#### L.sys

Das File L.sys enthält für jedes entfernte System, mit dem vom lokalen System aus eine Konversation begonnen werden kann, eine Zeile.

#### Format einer Zeile:

```
sys time dev speed phone login
```

Die Felder sind durch Leerzeichen voneinander getrennt.

## MUTOS 1700

### Bedeutung der Felder:

- sys - Name des entfernten Systems
- time - enthält die Wochentage und Zeiten, zu denen das entfernte System bereit ist, einen Ruf zu beantworten (z.B. MoTuTh0700-1630). "WK" bedeutet-jeder beliebige Arbeitstag und "Any"-jeder beliebige Tag. Da für eine Direktverbindung (diese Version von UUCP wurde nur für eine Direktverbindung implementiert) eine Zeitbeschränkung weniger sinnvoll ist, wird hier auf die Zeitangabe nicht näher eingegangen (ein Fehlen der Stunde-angabe bedeutet-jede beliebige Zeit).
- dev - Name des für die Verbindung zu diesem Rechner benutzten Gerätes. Anzugeben ist nur der letzte Teil des entsprechenden Special-Files (z. B. ifss).
- speed- Übertragungsgeschwindigkeit, die für den Datenaustausch auf dieser Leitung benutzt wird (möglich sind: 2400, 4800 und 9600 Bd.; Für AC A 7100 wird die Geschwindigkeit 2400 Bd. empfohlen).
- phone- Der Inhalt dieses Feldes muß mit dem Feld dev übereinstimmen.
- login- Dieses Feld, das Login-Informationen enthält, ist in eine Reihe von Unterfeldern gegliedert, die folgende Form haben:

```
expect send [expect send] ...
```

### Bedeutung:

- expect - Der Inhalt dieses Feldes wird beim Login-Prozeß (Anmelden am entfernten System) erwartet (gelesen) , z.B. der Ausdruck "login: ".
- send - Der Inhalt dieses Feldes wird an das entfernte System gesendet, falls expect erhalten wurde (z.B. der Login-Name zur Registration am entfernten System).

Das Feld expect kann wiederum als eine Folge von Unterfeldern der folgenden Form realisiert werden:

```
expect [-send-expect] ...
```

Hierbei wird send in dem Fall gesendet, wenn das vorhergehende expect nicht erhalten wurde. Das expect , das dem Feld send folgt, ist dann das nächste Wort, Diese Form ist zu empfehlen, da es wenig wahrscheinlich ist, daß das entfernte System gerade zu dem Zeitpunkt, zu dem ein Ruf dorthin vorbereitet wird (wovon das System noch nichts weiß) eedd auf die Leitung sendet. Ist der Inhalt des Feldes -send "BREAK", wird an das entfernte System das Zeichen "CTRL/Z" gesendet, wodurch dort ein neuer Login-Prozeß aktiviert wird (auf der Leitung erscheint "login: "). Der Code des zu sendenden Zeichens kann auch direkt in oktaler Form angegeben werden, z.B. '\004'. Eine solche Angabe kann bei einer Kopplung zu einem anderen Rechnertyp notwendig sein.

### Beispiel:

```
Labor1 Any ifss 2400 ifss login:-BREAK-login: uucp
```

## L-devices

Dieses File enthält für jedes Gerät, über das eine UUCP- Kopplung vorgesehen ist, je eine Zeile. Das Format ist folgendes:

```
type device call-unit speed
```

Bedeutung der Felder:

type - Typ der Verbindung ("DIR" für Direktverbindung)  
 device - Name des Gerätes, das die Koppelleitung bedient.  
 call-unit - Der Inhalt dieses Feldes muß dem von device entsprechen.  
 speed - Übertragungsgeschwindigkeit

Beispiel für eine Zeile:

```
DIR ifss ifss 2400
```

## SQFILE

UUCP führt im File SQFILE Zähler, die die Anzahl von erfolgreichen Verbindungen zu dem jeweiligen System enthalten. Beim Verbindungsaufbau wird der aktuelle Zählerstand vom MASTER (dem die Verbindung initialisierenden Rechner) an den SLAVE mit übertragen. Dort erfolgt ein Vergleich. Wenn der empfangene Zählerstand nicht mit dem eigenen übereinstimmt, so wird die Verbindung abgebrochen. Für jedes entfernte System existiert im File SQFILE eine Zeile.

Format:

```
sys count time
```

sys - Name des entfernten Systems  
 count - Anzahl der erfolgreichen Verbindungen zu diesem System  
 time - Zeitpunkt der letzten Übertragung

Dieses File muß ebenfalls vom Nutzer bei der Implementierung von UUCP errichtet werden. Dabei wird jedoch auf jeder Zeile nur der Name des entfernten Systems angegeben. Die beiden anderen Felder werden vom Programmpaket UUCP belegt und ständig aktualisiert.

### 2.2.3. Temporäre Files

~~~~~  
Außer SEQF, das sich in /usr/lib/uucp befindet, werden sämtlich hier beschriebenen temporären Files im Spool - Directory (/usr/spool/uucp) errichtet.

#### Temporäre Daten-Files

Während des Empfangs von Daten von einem entfernten Rechner werden die empfangenen Datensätze im Spool-Directory in temporären Daten-Files zwischengespeichert. Die Namen dieser Daten-Files haben folgendes Form:

TM.pid.nnn

Hiebei sind

TM - ein fester Präfix  
pid - eine Prozeßidentifikationsnummer  
nnn - eine dreistellige Nummer, die bei jedem Start von UUCICO bei Null beginnt und für jedes empfangene File um eins erhöht wird.

Nachdem das entfernte File vollständig empfangen wurde, wird das TM-File an den gewünschten Bestimmungsort kopiert, wobei es auch den gewünschten Namen erhält. Dabei wird das TM-File gelöscht. Nur wenn UUCICO anormal beendet wird oder während des Kopierens an den Zielort ein Abbruch erfolgt, bleibt das TM-FILE im Spool-Directory stehen. Ein regelmäßiges Löschen aller älteren TM-Files (mittels uuclean -pTM ist deshalb ratsam).

### Log-Files (LOGFILE, SYSLOG)

Während der Arbeit von UUCP, UUCICO, UUX und UUXQT werden Informationen über eingereichte Nutzeranforderungen (Kommandos), über erfolgreich oder nicht erfolgreich durchgeführte Rufe an ein entferntes System, über die Ausführung von UUX-Kommandos sowie über den Status von Kopieroperationen in das File LOGFILE im Spool-Directory eingetragen. Die möglichen Eintragungen sind in Anlage A der Anleitung für den Bediener erklärt. Unter bestimmten Umständen (parallele Arbeit mehrerer UUCP-Komponenten und gleichzeitiger gemeinsamer Zugriff zum LOGFILE) werden mehrere individuelle Log-Files gebildet, deren Namen die Form LOG.nnn hat. Diese individuellen Log-Files sollten mittels uulog in ein File-LOGFILE vereint werden. Es ist ratsam, das LOGFILE in regelmäßigen Zeitabständen zu löschen, da neue Eintragungen einfach angefügt werden und die schon vorhandenen bestehen bleiben, wodurch das LOGFILE stetig wächst und auch schwer überschaubar wird.

In das File SYSLOG wird für jede erfolgreiche Übertragung eine Zeile eingetragen, die die Uhrzeit der Übertragung, Namen des Nutzers und des entfernten Systems, Angaben zur ausgeführten Operation (Senden, Empfangen), die Anzahl der übertragenen Bytes und die dafür benötigte Zeit enthält.

### System Status Files

System Status Files werden von UUCICO im Spool-Directory errichtet.

Ihr Namen hat die Form

STST.sys wobei sys der Name des entfernten Systems ist,  
mit dem eine Kommunikation im Gange ist.

Diese Files enthalten Informationen über bestimmte Fehlerzustände, wie z.B. Fehler beim Login oder generell beim Verbindungsaufbau usw. Während des Datenaustausches zwischen zwei Rechnern enthält das File den Status "TALKING". Bei fehlerfreiem Verlauf der Konversation wird das File bei Abbruch der Verbindung wieder gelöscht. Bei anormaler Beendigung von UUCICO bleibt das File im Spool-Directory erhalten. Das hat zur Folge, daß während der durch RETRYTIME festgelegten Zeitspanne (wird im File /usr/src/cmd/uucp/uucp.h vor der Generierung festgelegt, im Standardfall 0) ein WiederholungsruF zu diesem Rechner nicht erlaubt wird. Wenn der Grund des vorzeitigen Abbruchs der Verbindung eine Nichtübereinstimmung der Sequenzzähler (im File SQFILE, siehe Punkt 2.2.2.) war, so muß das STST-File generell vor Start einer erneuten Übertragung zu diesem Rechner gelöscht werden, sonst erfolgt kein Ruf. Bei einer anormalen Beendigung des Programms UUCICO (z.B. durch ein kill oder einen Systemabsturz) kann es sein, daß das STST-File mit dem Status "TALKING" bestehen bleibt. In einem solchen Fall muß das File ebenfalls vor Beginn einer neuen Übertragung gelöscht werden.

### Lock - Files

Lock-Files werden für jedes aktive Gerät und für jedes entfernte System, mit dem eine Konversation läuft, gebildet.

Die Namen der Lock-Files haben folgende Form:

LCK.name

wobei name - der Name des Gerätes oder des entfernten Systems ist.

Lock-Files werden gebildet, um einen parallelen Zugriff (z.B. seitens mehrerer zu der gleichen Zeit aktiver UUCICO-Programme) auf ein und dasselbe Gerät bzw. entfernte System zu verhindern. Um zu sichern, daß der Zugriff mehrerer Komponenten zum gemeinsam genutzten LOGFILE streng sequentiell verläuft, wird von der jeweiligen Komponente vor dem Zugriff zum LOGFILE ebenfalls ein entsprechendes Lock-File errichtet (LCK.LOG) und sofort nach dem Zugriff wieder gelöscht. Gleiches gilt für das SQFILE (LCK.SQ). Falls das UUCICO-Programm anormal beendet wird, können Lock-Files im Spool-Directory verbleiben. Nach einer Zeit von 24 Stunden werden diese ignoriert (neu genutzt). Soll jedoch ein neuer Ruf erfolgen (zu dem gleichen System oder über das gleiche Gerät), müssen diese Lock-Files vorher gelöscht werden, sonst findet kein Ruf statt.

Files in /etc

Für die Realisierung der Gleichberechtigung aller an der Kopplung beteiligten Rechner ist es notwendig, daß das Programm UUCICO Zugriff zu bestimmten Systeminformationen hat (Prozeßidentifikation des Login-Prozeßes für jede Leitung und zugehöriger Gerätenamen). Diese Informationen werden von der Systemkomponente getty in Files im Directory /etc geschrieben, deren Namen folgende Form hat:

LOGPID.nnn

wobei nnn-die Nummer des Prozesses ist.

loginpid

uucico bildet aus den einzelnen Files /etc/LOGPID.nnn im Spool-Directory ein File "loginpid". Dieses File wird vom Programm UUCICO selbst aktualisiert. Von Seiten des Nutzers (Systemverantwortlichen) sind bezüglich dieser Files keine Operationen notwendig.

C. und D.-Files

Das Programm UUCP analysiert das gegebene Kommando und bereitet die Übertragung vor. Als Resultat dieser Vorbereitung errichtet UUCP im Spool-Directory C.- und D.-Files, die die auszuführende Operation näher beschreiben (C.\*) und die zu übertragenden Daten enthalten (D.\*). Das Format der Namen für diese Files ist folgendes:

C.sysnxxx                bzw.  
D.sysnxxx

Hierbei bedeuten

- sys - Name des entfernten Systems, mit dem kommuniziert werden soll,
- n - ein mittels UUCP-Option "-g" einfügbarer Buchstabe (Standard "n"), der zu einer besonderen Kennzeichnung benutzt werden kann.
- xxxx - Eine Nummer, die von UUCP bei jedem File erhöht wird.

Jedes C.-File besteht aus einer oder mehreren Zeilen, deren Inhalt aus folgenden Feldern besteht:

T File1 File2 User Options [D-file] [Mode]

Hierbei sind:

- T - Typ der auszuführenden Operation (S-Senden, R-Empfangen von Daten, X-Ausführen eines entfernten UUCP),
- File 1 - voller Pfadname des zu kopierenden Files
- File 2 - voller Pfadname des Zielortes
- user - Name des Nutzers, der die Übertragung angefordert hat
- Options - die Übertragung betreffende Options
- D-File - Namen des D-Files, das die zu übertragenden Daten enthält (nur bei TYP "S").
- Mode - Modus des D-Files

Das D-File enthält die zu übertragenden Daten. UUCP kopiert im Standardfall (ohne c-Option) das zu übertragende File in das Spool-Directory, wobei es einen Namen der Form D.sysnxxxx erhält (analog zum D-File). Nach erfolgreich ausgeführter Operation werden sowohl C.- als auch D.-Files gelöscht.

#### X.-Files

X.-Files, d.h. auszuführende Files werden von UUX im Spool-Directory erzeugt. Diese Files werden von UUXQT zur entfernten Kommandoausführung benutzt. Sie enthalten Informationen zum auszuführenden MUTOS-Kommando.

Das Format des Namens eines X.-Files ist folgendes:

X.sysnxxxx

Die Bedeutung der einzelnen Teile des Namens sind analog zu den beim C.-File beschriebenen. Ein X.-File besteht aus mehreren Zeilen, deren Anzahl von der Art des gewünschten Kommandos abhängt. Das erste Feld jeder Zeile (ein Buchstabe) kennzeichnet die inhaltliche Bedeutung der Zeile. Möglich sind folgende Kennzeichen:

#### -U Nutzer System

Nutzer - Nutzer, der die Kommandoausführung angewiesen hat  
System - Systemname des Nutzers

#### -F File-Name Ursprungsname

File-Name - generierter Name des Files, unter dem es beim Ausführungssystem gespeichert wird  
Ursprungsname - letzter Teil des Pfadnamens des Ursprungs-Files  
Es können keine oder mehrere Zeilen mit dem Präfix F enthalten sein (für jedes zur Kommandoausführung benötigte File eine Zeile).

## MUTOS 1700

### -I File-Name

File-Name - Standard-Eingabe für das auszuführende Kommando  
Die Standard-Eingabe wird entweder durch "<" in der uux-Kommandozeile spezifiziert oder über eine Pipe von der Standard-Eingabe des uux-Kommandos übernommen. (die uux-Option "-" wurde angegeben). Falls eine Standard-Eingabe nicht angegeben wurde (mit den oben beschriebenen 2 Möglichkeiten), wird "dev/null" benutzt.

### -O File-Name Systemname

File-Name - Standard-Ausgabe für das auszuführende Kommando  
Systemname - System auf dem die Standard-Ausgabe erfolgen soll.  
Die Standard-Ausgabe wird durch ">" in der uux-Kommandozeile spezifiziert. Wurde keine Standard-Ausgabe angegeben, wird "/dev/null" benutzt.

### -C Kommando [Argumente] ...

Kommando - auszuführendes Kommando  
Argumente - in der uux-Kommandozeile spezifizierte Argumente  
Die Standard-Eingabe und Standard-Ausgabe erscheinen in dieser Zeile nicht mit.

Das Format jeder Zeile hängt vom Kennzeichen am Zeilenbeginn ab. Nach erfolgreicher Kommandoausführung wird das X.-File wieder gelöscht.

## AUDIT

Dieses File wird im Spool-Directory des SLAVE-Systems errichtet, wenn der MASTER für die Übertragung das Debugging-Protokoll eingeschaltet hatte ("-x" Option bei uucico). Beim MASTER wird dieses Protokoll standardgemäß auf dem Bildschirm des Bedienterminals ausgegeben. Das Protokoll, das die Aktivitäten des SLAVE-Systems dokumentiert (wird beim Einschalten des Protokolls beim MASTER automatisch mit eingeschaltet), wird in das File AUDIT geschrieben und bleibt nach Beendigung der Übertragung erhalten.

## SSSEEEQQQFFF

Dieses File, das sich im Directory /usr/lib/uucp befindet, beinhaltet nur eine Zeile. Hier wird ein Zähler geführt, der beim Errichten jedes temporären C.-, D.- oder X.- Files im Spool-Directory um 1 erhöht wird. Dieser Zähler wird benutzt, um die Einmaligkeit jedes dieser Filenamen zu garantieren. Das Vorhandensein dieses Files ist nicht Bedingung. Bei nicht vorhandenem File SEQF wird es neu durch UUCP, UUCICO und UUXQT errichtet.

## 2.3. Arbeitsweise

### 2.3.1. Allgemeines

Jedes UUCP-System hat ein Spool-Directory, in welchem Anweisungen für auszuführende Kopplungen mit anderen Rechnern abgespeichert werden (Kommando-Files, zu übertragende Files, auszuführende MUTOS-Kommandos).

Die Programme UUCP und UUX, die das Hauptinterface für den Bediener bilden, analysieren das gegebene Kommando und schreiben alle für die Datenübertragung bzw. entfernte Kommandoausführung benötigten Informationen in das Spool-Directory.

Das Programm UUCICO realisiert die eigentliche Übertragung, die im Hintergrund stattfindet. Auf der Grundlage der aus dem Spool-Directory entnommenen Informationen wählt UUCICO die Übertragungsleitung aus und stellt die Verbindung zum gewünschten Rechner her. Dabei meldet sich das lokale UUCICO am entfernten System an (login), woraufhin dort ebenfalls UUCICO gestartet wird. Nachdem auf diese Weise der Kontakt zwischen beiden UUCICO-Programmen hergestellt wurde, stimmen diese sich über das zu benutzende Übertragungsprotokoll ab und führen die Datenübertragung aus. Zuerst überträgt das Programm, welches die Übertragung initialisierte, alle angewiesenen Daten (entsprechend den Aufträgen im eigenen Spool-Directory). Dann fragt es das andere Programm, ob bei diesem System ebenfalls Übertragungsaufträge vorliegen. Ist das der Fall, werden diese ausgeführt. Wenn auf keiner der beiden Seiten mehr Aufträge vorliegen, beenden beide UUCICO-Programme ihre Arbeit.

Wird eine entfernte Kommandoausführung angewiesen (UUX), startet das UUCICO-Programm des Zielrechners nach erfolgreicher Übertragung das Programm UUXQT, welches das gewünschte MUTOS-Kommando ausführt. Nach Beendigung der entfernten Kommandoausführung erfolgt eine mail-Mitteilung an den Nutzer, der den Auftrag ausgelöst hatte.

### 2.3.2. Vorbereitung der Übertragung

Zum Anweisen einer Datenübertragung steht dem Bediener das Kommando uucp zur Verfügung. Das Programm UUCP analysiert das gegebene Kommando und nimmt die notwendigen Eintragungen in das Spool-Directory vor.

UUCP führt eine Kontrolle der angegebenen Pfadnamen durch. Dabei wird zuerst der jeweilige Systemname geprüft. Der Systemname gilt dann als zulässig, wenn er auch im File /usr/lib/uucp/L.sys enthalten ist. Anderenfalls erfolgt eine entsprechende Ausschrift an den Bediener. Von den angegebenen Pfadnamen werden nun diejenigen weitergehend geprüft, deren Systemteil (Text bis "!") mit dem Namen des lokalen Systems übereinstimmt oder leer ist. Wurde nicht der volle Pfadname angegeben, sondern nur der letzte Teil, "dehnt" UUCP diesen Namen aus, d.h. es bildet den vollen Pfadnamen (vorangestellt wird der Pfad des aktuellen Directories). Daraufhin wird die Zugriffserlaubnis des Nutzers für diesen Pfadnamen geprüft (auf der Grundlage des USERFILE).

Ausgehend von den angegebenen Pfadnamen führt UUCP eine Klassifizierung der auszuführenden Operationen in 5 Typen durch:

- Typ 0 - Die Files werden am lokalen Rechner von ihrem Ausgangsort zum Bestimmungsort kopiert. UUCP führt dazu ein cp Kommando aus .

Typ 1 - Empfang von Files von einem entfernten Rechner. UUCP erstellt ein entsprechendes Arbeits-File (Präfix C, siehe 2.2.3.) im Spool-Directory (Typ der auszuführenden Operation "R").

Typ 2 - Senden von Files an ein entferntes System. Bei positiver Beendigung der oben genannten Tests der Zugriffsrechte werden die zu übertragenden Files in Form von Daten-Files (Präfix D) in das Spool-Directory kopiert (falls die -c-Option bei uucp nicht angegeben wurde). Ein entsprechendes Arbeits-File (Präfix C) wird ebenfalls erstellt (Typ der Operation "S").

Typ 3 - Übertragen von Files von einem entfernten System an ein anderes.

Typ 4 - Empfangen von Files von einem entfernten System, wenn die Pfadnamen der zu empfangenden Files Shell-Sonderzeichen (?,\*,[]) enthalten.

Bei Typ 3 und Typ 4 generiert UUCP ein uucp Kommando und sendet es an den entfernten Rechner zur Ausführung.

Nachdem alle Aufträge in das Spool-Directory eingetragen wurden (alle Tests erfolgreich), startet UUCP das Programm UUCICO, das diese Übertragungsaufträge ausführt.

### 2.3.3. Verbindungsaufbau

Der Verbindungsaufbau wird, wie auch die gesamte Übertragung von UUCICO realisiert. UUCICO kann auf verschiedene Weise gestartet werden:

- (a) - durch einen System-Daemon (zyklisch, um noch nicht abgearbeitete Aufträge auszuführen),
- (b) - Durch die Programme UUCP, UUX, UUXQT oder UUCICO,
- (c) - direkt durch den Bediener (vorwiegend für Testzwecke),
- (d) - durch ein entferntes System (Das Programm UUCICO muß dabei als Standard-Shell im UUCP-Login-Eintrag im File /etc/passwd angegeben worden sein.).

Wird UUCICO durch (a), (b) oder (c) gestartet, übernimmt es die MASTER-Rolle. Das bedeutet, es beginnt eine Verbindung mit einem entfernten System zu initialisieren.

Wenn UUCICO durch ein entferntes System gestartet wird (entsprechend (d)), arbeitet es im SLAVE-Modus (Die Verbindung wurde bereits vorher vom Partner -MASTER- hergestellt.).

Im MASTER-Modus prüft UUCICO zunächst das Spool-Directory auf vorhandene Arbeits-Files (C-Files). Wurde ein Arbeits-File gefunden, startet UUCICO einen Ruf an das darin spezifizierte System. Ist kein Arbeits-File vorhanden, beendet UUCICO seine Arbeit.

Wurde beim Start von UUCICO (im MASTER-Modus) die "-s" -Option angegeben, führt

UUCICO nur Arbeiten für das mit der "-s"-Option spezifizierte System aus. Dabei erfolgt in jedem Fall ein Ruf an dieses System, auch wenn kein Arbeits-File vorhanden ist (Damit können Aufträge für das eigene System, die im Spool-Directory des entfernten Rechners stehen, abgearbeitet werden.). Auf diese Art kann ein Polling eines entfernten Systems realisiert werden.

Für den Aufbau einer Verbindung zu einem entfernten System benutzt UUCICO die Daten des C-Files, sowie Informationen, die in verschiedenen Files im Directory /usr/lib/uucp enthalten sind (siehe Punkt 2.2.2. und 2.2.3.).

Zu Beginn des Rufes, nach dem Lesen des C-Files, wird im Spool-Directory ein Lock-File für das entfernte System errichtet, um eine mehrfache, parallele Kommunikation zwischen zwei Systemen zu verhindern.

Anhand des Systemnamens wird im File L.sys die diesem System zugeordnete Zeile gesucht. Der Inhalt des "time"-Feldes dieser Zeile wird nun mit der aktuellen Zeit verglichen um festzustellen, ob zu dieser Zeit ein Ruf erlaubt ist. Mit Hilfe der ebenfalls aus diesem File entnommenen Daten zu Gerätenamen und Übertragungsgeschwindigkeit sucht UUCICO im File L-devices ein für diesen Ruf geeignetes Gerät. Wenn ein Gerät erfolgreich "eröffnet" wurde (open), schreibt UUCICO wiederum ein entsprechendes Lock-File in das Spool-Directory, um zu verhindern, daß ein parallel arbeitendes UUCICO zum gleichen Gerät zugreift.

Nachdem die Auswahl und das Eröffnen des Gerätes erfolgreich abgeschlossen wurden, nutzt UUCICO die in L.sys enthaltenen Login-Informationen, um sich am entfernten System anzumelden (siehe Punkt 2.2.2.-L.sys). Verläuft das Anmelden erfolgreich, wird am entfernten System ebenfalls UUCICO gestartet (als Standard-Shell im für UUCP angelegten Eintrag im Password-File des entfernten Systems) und zwar im SLAVE-Modus.

Die Konversation zwischen den beiden UUCICO-Programmen beginnt mit einem Handshake, der von dem gerufenen (dem SLAVE) System gestartet wird.

Das SLAVE-System sendet dem MASTER (dem in MASTER-Modus gestarteten UUCICO) eine Nachricht, um ihm mitzuteilen, daß es bereit ist, die Anfangsnachricht zu empfangen. Daraufhin sendet der MASTER seinen Systemnamen und die Sequenznummer der Verbindung (siehe Punkt 2.2.2.-SQFILE). Diese Daten werden beim SLAVE auf Zulässigkeit geprüft. Wenn diese Prüfung positiv ausfällt, beginnt die Auswahl des Übertragungsprotokolls. Das SLAVE-System kann aber auch mit einer Nachricht "call-back ist gefordert" antworten (Das heißt, im USERFILE des SLAVE war für dieses System die call-back Option angegeben worden.). Daraufhin wird die laufende Verbindung beendet und der Verbindungsaufbau beginnt von neuem, nun jedoch von Seiten des ehemaligen SLAVE-Systems, das jetzt die MASTER-Rolle übernimmt. Damit erfolgt eine zusätzliche Verifikation des Partner-Systems.

#### Auswahl des Übertragungsprotokolls

Das entfernte System (SLAVE) sendet eine Liste der dort vorhandenen Übertragungsprotokolle (Im Standardfall gibt es nur ein Protokoll -den Paketdriver). Der andere Partner vergleicht diese Liste mit den im eigenen System vorhandenen Protokollen und wählt ein bei beiden vorhandenes aus. Wird kein gemeinsames Protokoll gefunden, erfolgt eine entsprechende Mitteilung und die Verbindung wird beendet.

Alle Nachrichten, die während der Phase des Verbindungsaufbaus ausgetauscht werden, sind zur Sicherung der Synchronisation in ein beginnendes DLE (Code 020) und in ein abschließendes NULL- Zeichen (Code 0) eingeschlossen.

Nach erfolgter Auswahl des Protokolles wird dieses gestartet und der Austausch aller folgenden Steuernachrichten und Daten wird unter Benutzung dieses Protokolls durchgeführt).

Der Nachrichtenaustausch während des Verbindungsaufbaus ist in Anlage A1

dargestellt.

#### 2.3.4. Datenaustausch

Die Rollen zu Beginn der Datenübertragung (MASTER oder SLAVE) entsprechen dem Modus mit dem jedes Programm gestartet wurde. Der Datenaustausch wird durch 2 Protokollebenen realisiert. Das ist erstens die Ebene, in der die Steuermitteilungen zur Steuerung des gesamten Prozesses des Austausches von Daten realisiert werden. Diese Ebene ist fest im Programm UUCICO verankert und wird durch die Protokollauswahl beim Verbindungsaufbau nicht beeinflusst. Die zweite darunterliegende Ebene ist die des Übertragungsprotokolls, das die fehlerfreie Übertragung der Steuermitteilungen der Ebene 1 und der Daten selbst absichert. Im Standardfall (nur ein Übertragungsprotokoll zur Auswahl) ist das der Paketdriver, der alle zu übertragenden Nachrichten in Pakete verpackt.

\_E\_b\_e\_n\_e\_l:

Zur Steuerung des Datenaustausches werden 5 Nachrichtentypen benutzt:

- S - der MASTER sendet ein File
- R - der MASTER empfängt ein File
- C - Kopieren des Files an den Zielort wurde beendet
- X - der SLAVE führt ein uuuuuuccppp - Kommando aus
- H - Beendigung der Übertragung.

Der MASTER sendet R, S oder X - Nachrichten (und die dazugehörigen Daten), bis alle Aufträge aus seinem Spool-Directory abgearbeitet sind. Danach wird eine H - Nachricht gesendet.

Der SLAVE antwortet entsprechend mit SY, SN, RY, wobei Y und N -Yes oder No bedeuten. Die Antworten des SLAVE auf S und R ( Y oder N ) hängen von den Zugriffsrechten der gewünschten Files (Directories) ab. Die Zugriffsrechte werden ausgehend vom USERFILE und vom File-Modus (Lese/Schreiberlaubnis) der gewünschten Files (Directories) geprüft.

Beim Empfang von Files werden die eingehenden Daten in ein temporäres Daten-File (TM...) im Spool-Directory geschrieben. Nach dem vollständigen Empfang eines Files wird es an den gewünschten Zielort kopiert. Danach sendet das empfangende System eine C - Nachricht an das sendende. Wurde das temporäre Daten-File erfolgreich an den Zielort kopiert, wird CY gesendet. Andernfalls (Fehler beim Kopieren, fehlende Zugriffsrechte, usw.) lautet die Nachricht CN, was bedeutet, daß das temporäre Daten-File in das von UUCICO errichtete Directory /usr/spool/uucppublic/user geschrieben wurde, wobei user der Name des Nutzers ist, der die Übertragung angewiesen hatte. In diesem Fall wird an dem System, das die Übertragung angefordert hatte, eine entsprechende mail - Nachricht an den Nutzer geschickt, unabhängig davon, ob die uucp - Option "-m" angegeben wurde oder nicht. (Eine mail - Nachricht wird auch bei allen anderen Ablehnungsfällen ( SN, RN, XN ) generiert.)

Wenn das SLAVE-System eine H - Nachricht erhält, durchsucht es das Spool-Directory nach vorhandenen Aufträgen. Liegen Übertragungsaufträge für das MASTER-System vor, antwortet der SLAVE mit HN . Daraufhin tauschen die beiden UUCICO-Programme die Rollen (MASTER <--> SLAVE) und die vorhandenen Aufträge werden abgearbeitet.

Findet der SLAVE in seinem Spool-Directory keine Arbeitseinträge antwortet er mit HY, was zum Abbruch der Verbindung führt.

Der Nachrichtenaustausch auf dieser Ebene ist in Anlage B dargestellt.

## Ebene 2 - Paketdriver:

~~~~~  
 Der Paketdriver ist ein Teil des Programms UUCICO, der aus den zu übertragenden Daten und Steuernachrichten (Ebene 1) Pakete formiert und diese mit einer Prüf-  
 folge versieht. Die Datenübertragung mit Hilfe des Paketdrivers ist vollkommen  
 transparent, d.h. es gibt keine Einschränkungen hinsichtlich der Codes der zu  
 übertragenden Daten.

Das Programm UUCICO kann auch mit einem Paketdriver arbeiten, der im Systemkern  
 enthalten ist (der Quellcode entspricht dem in UUCICO enthaltenen). Das wird  
 mit der "-g" -Option erreicht. Unter der aktuellen Version von MUTOS 1700 ist  
 dies jedoch nicht möglich, weil diese im Kern keinen Paketdriver enthält.  
 Der Paketdriver zerlegt die zu übertragenden Daten in Blöcke zu je 64 Byte (die  
 Steuermitteilungen werden entsprechend aufgefüllt). Jedem zu übertragenden  
 Paket wird ein Header von 6 Byte Länge vorangestellt. Vom Header wird eine  
 getrennte Prüfsumme gebildet.

Die Bytes des Header haben folgende Bedeutung:

- 1 - Synchronisationszeichen (DLE)
- 2 - kodierte Paketlänge
- 3,4 - Prüfsumme des Datenblockes
- 5 - Kontrollbyte (enthält Informationen über das  
 Paket)
- 6 - Prüfsumme des Headers (Exclusive - ODER der  
 Bytes 2-5)

Der Paketdriver wird durch eine spezielle Routine aktiviert. Die Paketdriver  
 der beiden kommunizierenden Systeme stellen dann eine logische Verbindung her,  
 wobei ein Austausch von Headern vollzogen wird. Verläuft das erfolgreich, sind  
 die Paketdriver beider Systeme bereit, alle ihnen übergebenen Informationen zu  
 übertragen.

Alle Pakete werden vom Sender fortlaufend nummeriert.

Der empfangende Paketdriver sendet dem sendenden für jedes empfangene Paket eine  
 Quittung. Die Zuordnung der Quittungen zu den gesendeten Paketen erfolgt über  
 die Paketnummer. Der Sender kann bis zu 3 Pakete im voraus senden, d.h. ohne  
 auf die entsprechenden Quittungen zu warten. Erst wenn auf 3 gesendete Pakete  
 keine Quittung vorliegt, unterbricht die sendende Seite die Übertragung und war-  
 tet auf Quittungen.

Bei Fehlersituationen (falsche Prüfsumme, falsche Headerprüfsumme, keine Antwort  
 in einer festgelegten Zeit (Timeout - 20s), ...) erfolgt eine Wiederholung der  
 Übertragung. Die Wiederholung beginnt beim ersten nicht quittierten Paket. Nach  
 5 aufeinanderfolgenden Fehlversuchen wird die Verbindung gelöst.

## 2.3.5. Verbindungsabbruch

~~~~~  
 Der Verbindungsabbruch erfolgt getrennt auf beiden Ebenen des Übertragungspro-  
 tokolls.

Zuerst wird die Verbindung der beiden Paketdriver beendet. Dabei tauschen die  
 beiden Seiten spezielle Ende-Nachrichten aus. Daraufhin werden alle belegten  
 Prüfsegmente freigegeben. Der anschließende Verbindungsabbruch auf der 1.,  
 höheren Ebene erfolgt ohne Paketdriver. (Die Nachrichten werden wieder in DLE  
 und NULL-Zeichen eingeschlossen und ohne Paketdriver übertragen.) Beide  
 UUCICO-Programme senden sich Ende-Nachrichten, die aus Nullen (Code 60)

bestehen. Sobald beide Seiten die Ende-Nachricht des Partners erhalten haben, gilt die Verbindung als gelöst.

Nach dem Lösen der Verbindung bereinigt das sich zu diesem Zeitpunkt im SLAVE-Modus befindliche Programm das Spool-Directory und beendet seine Arbeit. Das MASTER-Programm prüft erst sein Spool-Directory ob dort Aufträge für ein anderes System vorliegen (falls UUCICO nicht mit der "-s" -Option gestartet wurde). Ist das der Fall, beginnt der Verbindungsaufbau mit diesem System, sonst beendet das MASTER-UUCICO ebenfalls seine Arbeit.

## 2.4. Entfernte Kommandoausführung

~~~~~

### 2.4.1. Vorbereitung

~~~~~

Mittels des uux -Kommandos kann die Ausführung von MUTOS-Kommandos angewiesen werden, wenn der das Kommando ausführende Rechner und/oder einige der dazu benötigten Files entfernt sind.

UUX generiert entsprechend dem auszuführenden Kommando ein Ausführungs-File (Präfix X, siehe Punkt 2.2.3), das die Namen der benötigten Files (einschließlich Standard-Eingabe), den Login-Namen des Nutzers, den Zielort der Standard-Ausgabe sowie das auszuführende Kommando selbst enthält. Dieses File wird entweder in das lokale Spool-Directory geschrieben (wenn das Kommando auf dem lokalen Rechner ausgeführt werden soll) oder mittels eines Übertragungsauftrages (im Spool-Directory werden entsprechende C. und D. -Files erstellt) an das Ausführungssystem gesendet.

Für benötigte Files, die am Ausführungssystem nicht vorhanden sind, generiert UUX Empfangsaufträge ( C. -Files mit Arbeitstyp "R"). Diese Empfangsaufträge werden in das Spool-Directory des Ausführungssystems eingetragen.

Nach dem Errichten aller notwendigen Eintragungen im Spool-Directory ( C. -, D. - und X. -Files) startet UUX das Programm UUCICO, das diese Aufträge ausführt. Nachdem das UUCICO des Ausführungssystems alle Übertragungsaufträge erfolgreich abgearbeitet hat, startet es das Programm UUXQT, das das Ausführungs-File abarbeitet, d.h. das gewünschte MUTOS-Kommando ausführt .

### 2.4.2. Ausführung

~~~~~

Die Ausführung der durch uux angewiesenen MUTOS-Kommandos realisiert UUXQT. UUXQT wird entweder von UUCICO oder von UUX gestartet.

UUXQT prüft das Spool-Directory auf das Vorhandensein von Ausführungs-Files (Präfix X). Wurde ein Ausführungs-File gefunden, prüft UUXQT zunächst, ob alle zur Kommandoausführung benötigten Files vorhanden sind (anhand der mit "F" beginnenden Zeile im Ausführungs-File). Wenn noch nicht alle benötigten Files vorhanden sind, beendet UUXQT seine Arbeit mit diesem Ausführungs-File. Sind alle Files vorhanden, werden diese in das Directory /usr/lib/uucp/.XQTDIR kopiert, wo auch die eigentliche Kommandoausführung stattfindet.

Wurde für das auszuführende Kommando eine Standard-Eingabe oder eine Standard-Ausgabe spezifiziert (Zeilen im Ausführungs-File, die mit "I" bzw. mit "ö beginnen), werden diese eröffnet.

Nach erfolgreichem Verlauf dieser Vorbereitungen wird das auszuführende Kommando (mit "C" beginnende Zeile im Ausführungs-File) gestartet . Das erfolgt über den Start der SHELL mit der "-c" Option.

Wurde eine Standard-Ausgabe spezifiziert, kopiert UUXQT das Resultat der Kommandoausführung dorthin. Dabei sind 2 Fälle zu unterscheiden:

1. Befindet sich der Zielort auf dem das Kommando ausführenden System, erfolgt das Kopieren mittels eines cp - Kommandos.

2. Bezeichnet der Zielort ein entferntes System, bereitet UUXQT einen Übertragungsauftrag vor ( C. - und D. -Files im Spool-Directory) und startet UUCICO, das den Auftrag ausführt.

Bei der Ausführung von Kommandos, die einen fest definierten Ausgang haben (z.B. cc ), verbleibt das Resultat der Ausführung (z.B. File mit Objektcode) im Directory /usr/lib/uucp/.XQTDIR.

### 3. Anforderungen an Gerätetechnik und Betriebssystem

~~~~~

#### 3.1. Anforderungen an die Gerätetechnik

~~~~~

Die Nutzung der Kopplungskomponente UUCP setzt folgende Mindestkonfiguration des AC A 7100 voraus:

- AC A 7100 bzw. A 7150 mit 512 KByte HS
- 2 Minifloppy-Laufwerk doppelseitig, doppelte Dichte
- 1 ASP K 8071

Für die Verbindung der zu koppelnden Rechner werden der IFSS-bzw. der V.24-Kanal auf der ASP K 8071 benutzt. Bei der Nutzung der IFSS-Kanäle ist darauf zu achten, daß für die mit einer Verbindungsleitung gekoppelten Rechner die Strom-einspeisung für Sender und Empfänger auf der jeweiligen Anschlußsteuerung abgestimmt sein muß (Standard: Rechner1 -Sender und Empfänger aktiv, Rechner2 -Sender und Empfänger passiv). Bei Nutzung der V.24-Schnittstelle muß auf der ASP K 8071 die Einstellung asynchron erfolgen. Die Kopplung über die V.24-Schnittstelle ist nur für Duplexverbindungen vorgesehen (GDN K 8172, TAM 1200 oder vergleichbare Datenübertragungseinrichtungen, Direktverbindung).

#### 3.2. Anforderungen an das Betriebssystem

~~~~~

UUCP benötigt für seine Arbeit das Betriebssystem MUTOS 1700.

Die Anforderungen an das Betriebssystem sind minimal. Im Systemkern brauchen keinerlei Änderungen vorgenommen werden. Die notwendigen Zusätze, die in bestimmte, vom Betriebssystem genutzte Files eingetragen werden müssen, sind im Punkt 4.1. beschrieben.

## 4. Implementierung

~~~~~

### 4.1. Einbringen der notwendigen Zusätze in die Betriebssystemumgebung

~~~~~

Zunächst einmal muß dem IFSS-bzw. V.24-Kanal der ASP ein Gerätename zugewiesen werden (/etc/mknod). Die Eigenschaften dieses Gerätes müssen denen eines Terminals entsprechen.

Beispiele der Zuweisung:

```
cd /dev
/etc/mknod ifss c 4 0
/etc/mknod v24 c 6 0
```

Vorher sollte man sich vergewissern, daß der gewünschte Kanal nicht schon durch ein anderes Gerät logisch belegt ist (ls -l). Der Name des Gerätes ist beliebig. Es ist zu beachten, daß für jede Verbindungsleitung die entsprechenden IFSS-Kanäle der ASP elektisch abgestimmt sein müssen bzw. der V.24-Kanal auf "asynchron" eingestellt sein muß.

Weiterhin muß der Name des für die Kopplung benutzten Geräts in das File /etc/ttys eingebracht werden und zwar so, daß es beim Multiuseranlauf des Betriebssystems mit angesprochen wird (l am Beginn der Zeile). Als Terminaltyp ist für dieses Gerät der Typ "s" (satellite computer) anzugeben. Beispiel für eine Zeile in /etc/ttys:

```
lsifss
```

Anschließend ist ein Login-Eintrag für das Koppelprogramm UUCP im File /etc/passwd zu errichten. Als Login-Directory wird /usr/spool/uucppublic angegeben (falls dieser Standardpfadname in uucp.h benutzt wurde; sonst ist der entsprechende geänderte Pfadname anzugeben). Als Standard-Shell für diesen Eintrag fungiert /usr/lib/uucp/uucico.

Beispiel für eine entsprechende Zeile in /etc/passwd:

```
uucp::8:1::/usr/spool/uucppublic:/usr/lib/uucp/uucico
```

## 4.2. Benötigte Directories und Files

---

Die Generierung des Programmpakets UUCP erfolgt auf der Grundlage der Quellcodes aller zu diesem Paket gehörenden Teil- und Unterprogramme, sowie eines Makefiles, das die Verbindungen zwischen den einzelnen Teilprogrammen angibt. Im Standardfall befinden sich alle für die Generierung benötigten Files im Directory /usr/src/cmd/uucp. Dieser Pfadname ist jedoch nicht Bedingung. Im Ausgangszustand müssen sich in diesem Directory folgende Files befinden:

```

anlwrk.c, anyread.c, chkpth.c, chksum.c, cico.c,
cntrl.c, conn.c, cpmv.c, expfile.c, gname.c,
getargs.c, getprm.c, getpwinfo.c, gio.c, gnamef.c,
gnsys.c, gnxsq.c, gwd.c, imsg.c, index.c, ioctl.c,
lastpart.c, logent.c, makefile, pk0.c, pk1.c,
pkon.c, prefix.c, reserve.c, sdmail.c, shio.c,
sysacct.c, systat.c, ulockf.c, uuclean.c, uucp.c,
uucp.h, uucpname.c, uukill.c, uulog.c,
uuname.c, uurecover.c, uusecure.c, uux.c,
uuxqt.c, versys.c, xqt.c.
```

## 4.3. Generierung

---

### 4.3.1. Einbringen der nutzerspezifischen Parameter in den Quellcode

---

Vor der Compilierung der Quellprogramme müssen bestimmte Parameter entsprechend der konkreten Nutzeranforderungen modifiziert werden. Das betrifft Angaben zu den Namen der während der Compilierung sowie während der späteren Arbeit von UUCP benutzten Directories und zum Namen des lokalen UUCP-Systems.

änderungen im File uucp.h:

---

Im File uucp.h, das bei fast allen Quellcode-Files mit Hilfe der include-Anweisung mit eingebunden wird, sind folgende Definitionen zu ändern, falls die Standardwerte nicht benutzt werden sollen:

Definitionen für Directorynamen:

| Name der Variablen | Standardwert          |
|--------------------|-----------------------|
| THISDIR            | /usr/lib/uucp         |
| XQTDIR             | /usr/lib/uucp/.XQTDIR |
| SPOOL              | /usr/spool/uucp       |
| LOGDIR             | /usr/spool/uucp       |
| PUBDIR             | /usr/spool/uucppublic |

Definitionen für Filenamen:

| Name der Variablen | Standardwert            |
|--------------------|-------------------------|
| SQFILE             | /usr/lib/uucp/SQFILE    |
| SQTMP              | /usr/lib/uucp/SQTMP     |
| SEQFILE            | /usr/lib/uucp/SEQF      |
| SYSFILE            | /usr/lib/uucp/L.sys     |
| DEVFILE            | /usr/lib/uucp/L-devices |
| USERFILE           | /usr/lib/uucp/USERFILE  |
| SQLOCK             | /usr/spool/uucp/LCK.SQ  |
| SYSLOG             | /usr/spool/uucp/SYSLOG  |
| LOGLOCK            | /usr/spool/uucp/LCK.LOG |
| LOGFILE            | /usr/lib/uucp/LOGFILE   |

änderung des Namens des lokalen uucp-Systems:

|        |         |
|--------|---------|
| MYNAME | remotel |
|--------|---------|

Die einzelnen Directories und Files werden im Punkt 2.2. beschrieben.

änderungen der Zeit, nach der ein erneuter Verbindungsaufbau nach einem Fehlversuch gestartet werden darf:

|           |   |
|-----------|---|
| RETRYTIME | 0 |
|-----------|---|

änderungen im makefile:

Die folgenden änderungen sind dann notwendig, wenn nach der Compilierung aller Programme das Kommando make cp ausgeführt werden soll, das die neuerstellten Programme an den Zielort kopiert.

| Name der Variablen | Standardwert          |
|--------------------|-----------------------|
| DESTDIR            | /usr/lib/uucp         |
| OWNER              | uucp                  |
| INSDIR             | /usr/lib/uucp         |
| BIN                | /usr/bin              |
| PUBDIR             | /usr/spool/uucppublic |
| SPOOL              | /usr/spool/uucp       |
| XQTDIR             | /usr/lib/uucp/.XQTDIR |

Die Directory-Namen müssen den in uucp.h angegebenen entsprechen. Für BIN ist entweder /usr/bin oder /bin zulässig. Der Wert von OWNER (diesem Namen werden nach der Generierung sämtliche Files zugeeignet) muß gleich dem Login-Namen des für uucp angelegten Eintrages in Password-File (siehe 4.1.) sein.

#### 4.3.2. Compilerlauf

Nachdem alle notwendigen Änderungen in die Quellcode-Files eingebracht wurden, können diese übersetzt werden. Die Compilierung der Ausgangs-Files wird mit dem Kommando make gestartet. Die Systemkomponente make entnimmt dem File makefile alle notwendigen Verbindungen zwischen den Quell-Files und führt automatisch die Übersetzung aller Quell-Files sowie die Bildung der ausführbaren Files uucp, uucico, uux, uuxqt, uuclean, uulog, uuname und uukill durch.

#### 4.3.3. Bildung des Zielsystems

Der allgemeine Aufbau des Zielsystems, d.h. die einzelnen Directories und Files wurde in Punkt 2.2. beschrieben. Als Voraussetzung für die Bildung des Zielsystems werden die entsprechenden Directories errichtet:

```

/usr/lib/uucp
/usr/spool/uucp
/usr/spool/uucppublic
/usr/lib/uucp/.XQTDIR

```

Nachdem die Compilierung erfolgreich abgearbeitet wurde, d.h. die 4 primären und 4 sekundären Programme gebildet wurden, müssen diese in das Directory /usr/lib/uucp kopiert werden. Anschließend werden im Directory /usr/bin Links zu den sich in /usr/lib/uucp befindlichen Files uucp, uucico und uux errichtet (Kommando ln). Danach werden die von uucp benötigten Files erstellt (in /usr/lib/uucp). Der Inhalt dieser Files muß den Charakteristiken des konkreten Anwendungsfalls entsprechen (Namen der die Verbindungsleitungen bedienenden Geräte, Übertragungsgeschwindigkeit auf jeder Leitung, Namen der anzusprechenden entfernten Systeme, Loginnamen ...). Für das Errichten dieser Files ist es zweckmäßig Punkt 2.2. diesen Materials zu Hilfe zu nehmen, wo die inhaltliche Bedeutung sowie das Format der einzelnen Files detailliert beschrieben sind. Es wird empfohlen, daß Eigentümer und File-Modus der oben genannten Files und Directories wie folgt zugewiesen werden:

Die ausführbaren Programme uucp, uucico, uux und uuxqt, wie auch alle oben genannten Directories sollten als Eigentümer den Namen des für die Kopplung errichteten Login-Eintrags im Password-File (siehe Punkt 4.3.) besitzen. Für die Files sollte der File-Modus 4111 sein, d.h. erlaubt ist nur die Ausführung (das "setuid-bit" ist gesetzt). Damit ist es außenstehenden Personen nicht mehr möglich, sich durch bestimmte Manipulationen eine Standard-Shell für den uucp-Login-Eintrag zu errichten. Der File-Modus für die Files L.sys, L-devices, SQFILE und USERFILE hängt vom gewünschten Sicherheitsgrad des Systems ab. Der Eigentümer sollte jedoch in jedem Fall der Name des uucp-Login-Eintrags sein. Als Standard ist der Modus 0400 vorgesehen.

Die Directories /usr/lib/uucp/.XQTDIR und /usr/spool/uucp müssen den Modus 0777 aufweisen.

## MUTOS 1700

Die oben genannten Tätigkeiten können am einfachsten mit Hilfe von `make` ausgeführt werden.

Dazu gibt es folgende Möglichkeiten:

### `make makedirs`

Errichten der für `uucp` benötigten Directories, einschließlich Setzen von Eigentümer (laut `makefile`) und Modi.

### `make cp`

übersetzen der Quellmoduln (falls noch nicht realisiert) und Kopieren der ausführbaren Files (4 primäre und 4 sekundäre Programme) an den für die Ausführung vorgesehenen Zielort (z.B. `/usr/lib/uucp`, `usr/bin`) (einschließlich Setzen von Eigentümer und Modi).

### `make cpfiles`

Kopieren der Files `L.sys`, `L-devices`, `SQFILE` und `USERFILE` an den Zielort (`/usr/lib/uucp`). In diese Files müssen noch die anwenderspezifischen Parameter eingetragen werden.

### `make install`

Dieses Kommando ist eine Zusammenfassung der 3 oben genannten Kommandos. Es wird eine generelle Neuübersetzung aller Moduln sowie das Kopieren aller notwendigen Komponenten an den Zielort angewiesen.

## 5. Administrative Arbeit während der Nutzung

~~~~~

Während der Nutzung des Programmpaketes UUCP ist es notwendig, regelmäßig bestimmte Tätigkeiten auszuführen, die das Säubern der Directories sowie das Aktualisieren der einzelnen Files betreffen.

### Einbringen neuer Systeme

Falls eine Kopplung mit einem neuen, dem lokalen UUCP-System noch nicht bekannten entfernten Rechner gewünscht wird, müssen in den entsprechenden Daten-Files (die die Konfiguration bestimmen) folgende Eintragungen vorgenommen werden (Siehe dazu auch Punkt 2.2.2.).

- In das USERFILE muß für das neue System eine Zeile eingetragen werden, die die Zugriffsrechte festlegt.
  
- Im File L.sys muß eine Zeile für das System angefügt werden.
  
- Falls eine neue Leitung benutzt wird, ist diese in das File L-devices einzutragen. Die Leitung muß auch im File /etc/ttys (siehe Punkt 4.1.) vermerkt sein. Ein entsprechender Eintrag im Directory /dev ist Voraussetzung (siehe Punkt 4.1.).
  
- Das File SQFILE (siehe Punkt 2.2.3.) muß ebenfalls eine Zeile für das neue System enthalten.

### Änderung der Übertragungsgeschwindigkeit

Eine Änderung der Übertragungsgeschwindigkeit auf einer bestimmten Leitung erfolgt durch eine Änderung der entsprechenden Werte in den Files L.sys und L-devices. Möglich sind folgende Stufen: 2400 Bd, 4800 Bd.

Es wird empfohlen, die Standardgeschwindigkeit 2400 Bd zu nutzen, da der Nettodurchsatz nur begrenzt von der Übertragungsgeschwindigkeit auf der Leitung abhängt. Für die Stufen von 2400 und 4800 Bd ist etwa der gleiche Nettodurchsatz zu verzeichnen (belastungsabhängig zwischen 130 und 210 Zeichen pro Sekunde).

### Hinweis:

Die somit ausgewählte Geschwindigkeit muß mit der für die konkrete Leitung in getty eingestellten Geschwindigkeit (in der Tabelle für den Terminaltyp "s") übereinstimmen (Standardeinstellung 2400 Bd.) Die Einstellung über das Kommando stty ist für UUCP nicht ausreichend. Weiterhin müssen natürlich die Geschwindigkeiten von 2 miteinander kommunizierenden Rechnern übereinstimmen.

### Bereinigung der Directories

Das Spool-Directory ist in regelmäßigen Zeitabständen auf alte, nicht mehr benötigte Files zu prüfen, um eine Übersichtlichkeit dieses Directories zu wahren (Arten der Files - siehe Punkt 2.2.3.). Das betrifft auch das LOGFILE. Neue Eintragungen werden an den bestehenden Inhalt hinten angefügt. Deshalb ist es angebracht, das LOGFILE von Zeit zu Zeit zu löschen. Sind partielle LOGFILES entstanden (durch parallele Arbeit mehrerer UUCP) können diese durch uulog (siehe entsprechenden Abschnitt der Programmtechnischen Beschreibung, Teil 1 von MUTOS 1700) in ein LOGFILE vereinigt werden. Das Löschen nicht mehr benötigter Files kann bequem mittels uuclean (Angabe des Alters der zu löschenden Files möglich), durch eine selbst erstellte Shell-Prozedur oder einfach durch Benutzung der entsprechenden Kommandos des Betriebssystems MUTOS 1700 erfolgen.

Hinweis:

- LCK. / -Files, die wegen einer abnormalen Beendigung von UUCICO im Spool-Directory verblieben sind, verhindern eine erneute Übertragung zum gleichen System (über die gleiche Leitung). Sie müssen vor Anweisung einer neuen Übertragung gelöscht werden.
- Die im File SQFILE (siehe Punkt 2.2.2.) gespeicherten Sequenzzähler dürfen nicht willkürlich verändert werden (sonst kein erfolgreicher Verbindungsaufbau möglich - BADSEQUENCE). Diese Sequenzzähler müssen stets mit den analogen Zählern in den entsprechenden Partnersystemen übereinstimmen.

Da nach Übertragungen, die mit Fehlerstatus beendet wurden, die Aufträge im Spool-Directory verbleiben, ist es ratsam, periodisch UUCICO zu starten, wodurch diese Aufträge abgearbeitet werden.

## Anlage A Abläufe während des Verbindungsaufbaus

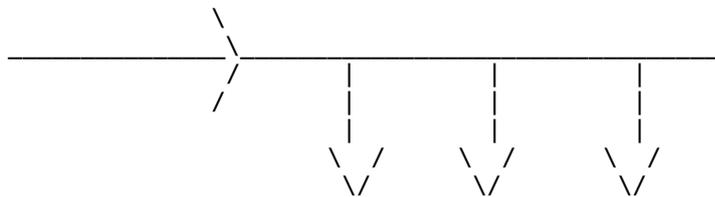
---

### Erklärung

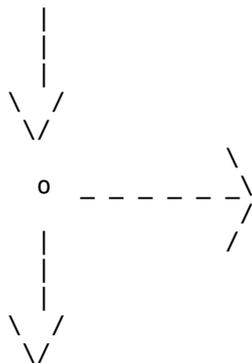
Beim Lesen der Steuermitteilungen wird in jedem Fall erst auf das beginnende DLE gewartet, d.h. das Leseprogramm verharrt so lange in einer Warteschleife, bis DLE empfangen wurde. Die nach DLE folgenden Zeichen bis zum NULL -Zeichen werden dann als eigentliche Mitteilung gelesen und ausgewertet.

"goto next" bedeutet, daß der Verbindungsaufbau mit dem entsprechenden System abgebrochen wird (goto next system). Das Programm UUCICO kehrt in den Ausgangszustand zurück und untersucht das Spool-Directory, ob noch Aufträge für ein anderes System vorliegen. Wenn ja, wird ein Verbindungsaufbau mit dem anderen System begonnen. Wenn keine C. -Files für andere Systeme mehr vorliegen, beendet UUCICO seine Arbeit.

Eine Abbildung der Form:



bedeutet, daß nur eine der nachfolgenden Möglichkeiten zutrifft, während



bedeutet, daß eine logische Steuerung in beiden Richtungen weitergeleitet wird.

- (1) - BREAK besteht aus einem Zeichen-CTRL/Z
- (2) - "uucp" steht hier für den Login-Namen, mit dem der MASTER sich beim SLAVE anmeldet. Bei ordnungsgemäßem Verlauf des Anmeldens wird am SLAVE UUCICO gestartet.
- (3) - Der MASTER teilt dem SLAVE seinen Namen, die Sequenznummer der Übertragung und die die Übertragung charakterisierenden Options mit.  
Myname-Name des UUCP-Systems, in uucp.h festgelegt

seq-Sequenznummer der Verbindung

- (4) - Abbruch des Verbindungsaufbaus, da der Übertragungskanal belegt ist (LCK-File steht im Spool-Directory).
- (5) - Abbruch des Verbindungsaufbaus, da die vom MASTER erhaltene Sequenznummer nicht mit der im eigenen System (SQFILE) für das MASTER-System gespeicherten übereinstimmt.
- (6) - Im File L.sys des SLAVE wurde für das MASTER-System die Option "call-back" spezifiziert. Dabei wird der Verbindungsaufbau abgebrochen und der SLAVE "ruft zurück", d.h. der Verbindungsaufbau beginnt von vorn, wobei der SLAVE-Rechner jetzt die MASTER-Rolle übernimmt.  
"xuucicö-Start von UUCICO im MASTER-Modus.
- (7) - Die empfangene Sequenznummer stimmt mit der im eigenen System gespeicherten überein. Fortsetzung des Verbindungsaufbaus.
- (8) - "ROK" wurde erhalten, d.h. der Anfangs-Handshake ist erfolgreich verlaufen. Nun wird die Liste der möglichen Protokolle erwartet (Mitteilung, die mit "P" beginnt).
- (9) - Der MASTER hat ein in beiden Systemen vorhandenes Protokoll ausgewählt und schickt nun die Identifikation dieses zu benutzenden Protokolls an den SLAVE (hier "g").  
"ü bedeutet use protocol.
- (10) - Es wurde kein Protokoll gefunden, das sowohl im SLAVE, als auch im MASTER vorhanden ist. Die Verbindung wird abgebrochen.
- (11) - Beim MASTER ist ein Fehler beim Initialisieren des ausgewählten Protokolls aufgetreten. Abbruch der Verbindung.
- (12) - Beim SLAVE ist ein Fehler beim Initialisieren des ausgewählten Protokolls aufgetreten. Abbruch der Verbindung.

## Anlage B Abläufe während des Datenaustausches

---

Die hier gezeigten Abläufe entsprechen der Ebene 1 des Datenaustauschprotokolls. Die darunterliegende Ebene des Paketdrivers findet in dieser Abbildung keinen Niederschlag.

Zu beachten ist, daß im Standardfall (alle hier gezeigten und Daten werden mit Hilfe des Paketdrivers übertragen) die einleitenden (und bei den Datenblöcken auch abschließenden) DLE-Zeichen entfallen. Diese DLE-Zeichen werden nur für den Fall benutzt, wenn ohne Paketdriver gearbeitet wird (uucico- Option "-g"). Dies ist jedoch für die Praxis nicht sinnvoll, da somit keine Datensicherung mehr erfolgt.

Die Steuermitteilungen werden beim Senden auf ein Vielfaches von 64 Byte aufgefüllt. Das letzte Zeichen muß dabei ein NULL-Zeichen sein.

Die Angabe der Richtung der Datenübertragung- "-S" Senden und "-R" Empfangen -bezieht sich stets auf den MASTER.

- (1) Im Spool-Directory Slave sind keine Aufträge für das MASTER-System vorhanden.
- (2) Im Spool-Directory des SLAVE befinden sich Übertragungsaufträge bezüglich des MASTER-Systems. Die beiden Systeme tauschen die Rollen (MASTER<-->SLAVE) und die Aufträge werden abgearbeitet.
- (3) MASTER übernimmt die SLAVE-Rolle. Die Übertragungsprozedur beginnt von vorn.
- (4) Erfolgreicher Abschluß des Datenaustauschs, Ausschalten des Übertragungsprotokolls (Paketdriver), Abbruch der Verbindung (verläuft auf Ebene 1).
- (5) Fehler in der empfangenen Kommandozeile oder Fehler beim Prüfen der Zugriffsrechte.
- (6) Start von UUCP im SLAVE mit den empfangenen Argumenten ist erfolgt.
- (7) Das zweite Zeichen der Antwort wird nicht ausgewertet (Y oder N). Den Start von UUCP im anderen Rechner kann nur der MASTER beim SLAVE veranlassen, nicht umgekehrt.
- (8) Zugriffsrechte zu dem vom MASTER aufgeforderten File nicht in Ordnung oder das File läßt sich nicht öffnen.
- (9) Fehler beim Schreiben der Daten.
- (10) Fehler beim Lesen der Daten.
- (11) Fehler beim Kopieren des gelesenen Files an den Zielort (Zugriffsrechte).
- (12) Das gelesene File wurde erfolgreich an den Zielort kopiert.
- (13) Das nach C folgende Zeichen wird nicht ausgewertet. Es wird nur für mail benutzt.

- (14) Unzulässiger Pfadname für den Zielort des zu empfangenden Files (Zugriffsrechte nicht vorhanden oder Directory nicht vorhanden bei Nichtangabe der "-d"-Option von UUCP (make directories)).
- (15) Das temporäre D-File, wohin die zu empfangenden Daten zuerst gelangen, läßt sich nicht öffnen.
- (16) Fehler beim Schreiben der Daten.
- (17) Fehler beim Lesen der Daten.
- (18) Fehler beim Kopieren der gelesenen Daten an den Zielort (Zugriffsrechte). Turnoff- bedeutet Abbruch der Datenaustauschprozedur, Ausschalten des Paketdrivers, Abbruch der Verbindung.