

# Anleitung

## IO-Mailer

DE 1.2 08/2001 FT

Konfiguration und  
Programmierung

Rel. 1.2, Aug. 2001

Typen:

57111, 57211

57511, 57611

**W&T**

**Inhalt**

<b>1</b>	<b>Steckverbinder .....</b>	<b>5</b>
<b>2</b>	<b>Inbetriebnahme .....</b>	<b>6</b>
2.1	IP-Adresse, Gateway und Subnet- Mask via DHCP .....	6
2.2	IP-Adresse, Gateway und Subnet-Mask ohne DHCP .....	7
2.2.1	IP-Adresse .....	7
2.2.2	Gateway und Subnet-Mask .....	7
2.3	Vorbereitung zum Mail-Empfang .....	7
<b>3</b>	<b>Mail-Kommandos .....</b>	<b>9</b>
<b>4</b>	<b>Konfiguration.....</b>	<b>11</b>
4.1	XML .....	11
4.1.1	Wieso XML? .....	11
4.1.2	Unterschiede zu HTML .....	12
4.1.3	Arbeiten mit Dokumentvorlagen .....	12
4.1.4	Die Unterteilung in Hauptzweige .....	16
4.1.5	Zeitangaben .....	18
4.2	Netzwerkumgebung (servers) .....	18
4.3	Postfachzugang (pop-account) .....	20
4.4	Absenderadresse des Mailers (device-id) .....	21
4.5	Erscheinungsbild des Gerätestatus (report) .....	22
4.6	Zeitzone (zone) .....	25
4.7	Zugriffsbeschränkungen (access) .....	26
4.8	Zeitabhängige Steuerung (schedule) .....	28
4.9	Die serielle Schnittstelle (serial) .....	29
4.10	Ereignisse und Aktionen (alarm) .....	31
-	Variablen .....	32
-	Operatoren .....	33
-	Triggerbedingungen .....	33
-	Serielle Triggerbedingungen .....	34
-	Aktionen .....	34
-	Empfängerlisten .....	34
-	Ereignisse .....	35
4.10.1	Einfache E/A-Beispiele .....	36
4.10.2	Beispiele zur Benutzung des seriellen Ports .....	38
<b>5</b>	<b>Einschränkungen .....</b>	<b>42</b>
5.1	Was nicht geht .....	42
5.2	Grenzwerte .....	42
5.3	Diverse mögliche Probleme .....	43
5.3	IO-Mailer und Exchange 5.5 .....	43

- 6 Internet-Adressen ..... 45**
  - 6.1 Zum Thema XML ..... 45
  - 6.2 Zum Thema Timeserver ..... 45
  
- 7 Firmware-Update ..... 47**
  - 7.1 Update des TCP/IP-Ethernet-Bereichs ..... 47
    - 7.1.1 unter Windows 9x/NT/2000 ..... 47
    - 7.1.2 unter UNIX ..... 48
  - 7.2 Update des SMTP/POP3-Cient-Bereichs ..... 50
    - 7.2.1 Update über das Netzwerk ..... 50
    - 7.2.2 Update über das Netzwerk ..... 51
  
- Technische Daten ..... 52**
  - Typen 57111, 57211 ..... 50
  - Typen 57111, 57211 ..... 53
  
- EG-Konformitätserklärung ..... 54**
  
- Anhang: Optionales Zubehör ..... 55**
  - Schraubklemmadapter Typ 50883 ..... 55
  - Beschaltungsbeispiel ..... 55

© 08/2001 by Wiesemann & Theis GmbH

2. überarbeitete Auflage

Irrtum und Änderung vorbehalten:

Da wir Fehler machen können, darf keine unserer Aussagen ungeprüft verwendet werden. Bitte melden Sie uns alle Ihnen bekannt gewordenen Irrtümer oder Mißverständlichkeiten, damit wir diese so schnell wie möglich erkennen und beseitigen können.

**1 Steckverbinder**

Der serielle Port und die digitalen Ein- und Ausgänge des IO-Mailers liegen an einer DB9 Buchse mit folgender Pinbelegung:

Pin	Funktion	Beschreibung
1	-	
2	TxD	Daten Ausgang
3	RxD	Daten Eingang
4	X0	bin. Eingang
5	GND	0V
6	Y0	bin. Ausgang
7	X1	bin. Eingang
8	Y1	bin. Ausgang
9	VCC	+5V

Wobei also die Reihenfolge der seriellen Leitungen RxD/TxD gegenüber der DTE-Belegung, wie sie z.B. am Com-Port eines PCs zu finden ist, vertauscht ist und damit DCE-Belegung entspricht. Zum provisorischen Anschluß eines seriellen Terminals kann gefahrlos (d.h. ohne zulässige Eingangsspannungen zu überschreiten oder Ausgänge gegeneinander kurzzuschließen) ein 1:1 belegtes seriellles Kabel dienen, wie man es auch zum Anschluß eines Modems verwenden würde.

Die binären Eingänge X0, X1 sind spannungsfest bis +/-30V, ihre Schaltschwelle liegt zwischen +1V und +4V, offene Eingänge werden als binäre 0 gelesen. Y0, Y1 sind Open Collector Ausgänge und können max. +30V und bis zu 100mA schalten. Die Versorgungsspannung VCC kann als Pull-Up-Spannung für die binären Eingänge verwendet werden und darüber hinaus kleine Ausgangslasten bis insgesamt 20mA versorgen.

Für die einfache Anschaltung der Ein- und Ausgänge kann auch der optional verfügbare Schraubklemmadapter Typ 50883 verwendet werden.

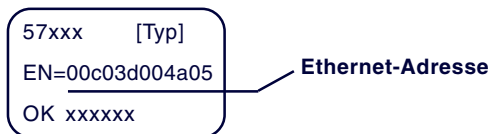
## 2 Inbetriebnahme

Für jeden Mailer, den Sie betreiben wollen, benötigen Sie eine *IP-Adresse*, die innerhalb Ihres Netzwerkes eindeutig sein muß, und der über DNS ein Hostname zugeordnet sein sollte. Außerdem benötigt der IO-Mailer eine eigene *Mail-Adresse*, und auf Ihrem Mail-Server sollte ein entsprechendes Postfach neu angelegt werden. („Sollte“ heißt dabei: Zumindest probenhalber ist es auch möglich, den IO-Mailer zu betreiben, ohne dass er Mail empfangen kann oder überhaupt eine gültige Absenderadresse hat. Wenn er aber Mail empfangen und verarbeiten soll, benötigt er auf jeden Fall ein eigenes Postfach und darf auf keinen Fall ein bereits vorhandenes Postfach mitbenutzen!)

*Was den organisatorischen Aufwand angeht, können Sie sich den IO-Mailer ungefähr wie einen neuen Mitarbeiter vorstellen, der als Besonderheit zwar einen eigenen Arbeitsplatzrechner mitbringt, der aber immer noch in Ihr Netzwerk und Ihr Mail-System eingebunden werden muß.*

### 2.1 IP-Adresse, Gateway und Subnet-Mask via DHCP einstellen

In Netzwerkumgebungen mit DHCP-Unterstützung, kann der IO-Mailer seine IP-Adresse, Gateway und Subnet-Mask vom DHCP-Server beziehen. Dazu muß die zu vergebende IP-Adresse durch eine feste Bindung an die Ethernet-Adresse des IO-Mailers reserviert werden. Unter Windows NT erfolgt dieses im DHCP-Manager unter dem Menüpunkt „Reservierungen“. Linux stellt zu diesem Zweck die Datei „DHCPconf“ zur Verfügung in die ein entsprechender Eintrag vorgenommen werden kann. Die Ethernet-Adresse des IO-Mailers finden Sie auf dem am Gehäuse befindlichen Aufkleber.



Die Übermittlung weiterer Parameter oder einer Lease-Time ist nicht möglich.

# W&T

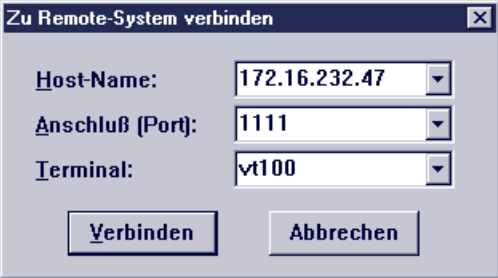
## 2.2 IP-Adresse, Gateway und Subnet Mask ohne DHCP-Unterstützung einstellen

### 2.2.1 IP-Adresse

Verbinden Sie sich über ein serielles Terminal mit dem IO-Mailer, Schnittstellenparameter 9600 Baud, 8 Datenbit, kein Handshake, und schließen Sie dann den Mailer an seine Spannungsversorgung an. Während er startet, tippen Sie am Terminal mindestens dreimal den Buchstaben „x“, bis die Eingabeaufforderung „IP no.+<ENTER>:“ erscheint. Geben Sie dann die vorgesehene IP-Adresse ein (vier Zahlen zwischen 0 und 255, durch Punkte getrennt). Bei Fehlern in der Eingabe antwortet das Gerät mit „FAIL“. Schalten Sie in dem Fall den IO-Mailer wieder aus, und versuchen Sie es noch einmal von vorne.

### 2.2.2 Gateway und Subnet-Mask

Diese Angaben sind nur erforderlich, wenn nicht alle benötigten Server (welche das sind s.u., Abschnitt „Netzwerkumgebung“) im selben Netzsegment wie der IO-Mailer selber zu finden sind. Fragen Sie im Zweifelsfall bitte Ihren Netzwerkadministrator. Um die entsprechenden Daten einzutragen, verbinden Sie sich über Telnet mit dem IO-Mailer, TCP-Portnummer 1111, und wählen Sie das Untermenü „SET-UP System“/“Setup TCP/IP“.



The image shows a dialog box titled "Zu Remote-System verbinden" with a close button in the top right corner. It contains three input fields, each with a dropdown arrow on the right: "Host-Name" containing "172.16.232.47", "Anschluß (Port)" containing "1111", and "Terminal" containing "vt100". Below these fields are two buttons: "Verbinden" and "Abbrechen".

## 2.3 Vorbereitung zum Mail-Empfang

Theoretisch lassen sich alle weiteren Betriebsparameter per e-Mail konfigurieren. Einige davon müssen allerdings bereits korrekt eingestellt sein, bevor der IO-Mailer überhaupt erst-

mals Mail empfangen kann. Das geht über das serielle Terminal, entweder in einem kleinen Frage- und Antwort-Dialog, wie im folgenden beschrieben. Oder Sie können auch gleich eine komplette Konfigurationsdatei erstellen (mehr dazu in Abschnitt 4) und diese per Terminal in den IO-Mailer einspielen. Wenn ihr Terminal das Senden vorbereiteter Textdateien überhaupt erlaubt, ist das vielleicht sogar die einfachere Lösung.

Verbinden Sie sich also nochmals per Terminal mit dem IO-Mailer, 9600 Baud, 8 Bit, Software-Handshake. Drücken Sie einmal die Leertaste, um das Tatstaturmenü anzuzeigen, und wählen Sie die Option „setup (by interview)“. Die Server für SMTP und POP3 können Sie entweder direkt als IP-Adressen oder über Hostnamen angeben, bei Hostnamen werden Sie aber zusätzlich noch nach der IP-Adresse eines DNS-Servers gefragt. Mit „Mail-Adresse des Administrators“ ist die Adresse gemeint, von der aus Sie die übrigen Konfigurationsdaten verschicken wollen. Sie können hier zunächst auch einfach ein „\*“ eingeben, sollten das in der endgültigen Konfiguration aber unbedingt noch korrigieren (s. u., Abschnitt „Zugriffsbeschränkungen“).

Um die Konfiguration zu testen, schicken Sie dem IO-Mailer eine Mail mit dem Betreff „help“. Wenn alles korrekt eingestellt ist, wird er die Mail innerhalb einer Minute abholen und beantworten. Am seriellen Terminal können Sie diesen Vorgang im Detail beobachten, bzw. bei Fehlschlag erhalten Sie dort auch Hinweise auf die mögliche Ursache des Problems.



## 3 Mail-Kommandos

Der IO-Mailer reagiert nur auf Mails, deren Betreffzeile mit einem gültigen Schlüsselwort beginnt. Gültige Schlüsselwörter sind (wobei Groß-/Kleinschreibung nicht beachtet wird):

`help:`

Kurze Zusammenfassung der verfügbaren Kommandos.

`unsubscribe:`

Erlaubt es den Empfängern automatisch erzeugter Mails, ihre eigene Adresse aus den Konfigurationsdaten zu streichen. Dies ist hauptsächlich als möglichst unkomplizierte Lösung für Fälle gedacht, in denen die Einrichtung dieser Benachrichtigungen mit einem Empfänger nicht abgesprochen war, und sollte bei sorgfältig vorbereitetem Einsatz des IO-Mailers nicht nötig werden.

`resume:`

Macht die Wirkung von „unsubscribe“ rückgängig.

`status:`

Zeigt den Zustand der Ein- und Ausgänge und Zählvariablen an.

`set:`

Wertzuweisung an eine oder mehrere Gerätevariablen. Kann zur Direktsteuerung der binären Ausgänge verwendet werden, z.B. „set y0 = 1“ oder „set y0=0 y1=1“, aber auch, um die internen Zähler zurückzusetzen, z.B. „set n0=0“, und zum Senden serieller Daten, z.B. „set tx=Alles klar?\r\n“.

`info:`

Liefert die aktuelle Gerätekonfiguration, in dem XML-Format, das auch zum Einstellen der Konfiguration verwendet wird. Um nur Teile der Konfiguration anzuzeigen, können Sie zusätzlich die Namen von einem oder mehreren der neun Hauptzweige angeben, z.B. „info alarm“ oder „info servers, access“.

## W&T

### template:

Ähnlich wie „info“, leere Felder in den Konfigurationsdaten werden dabei aber nicht weggelassen. Das ist weniger übersichtlich, macht es aber leichter, fehlende Informationen zu ergänzen.

### sample:

Liefert eine Beispiel-Konfigurationsdatei mit Triggerbedingungen, durch die bei Anlegen von Signal X0 bzw. X1 jeweils eine Mail-Benachrichtigung erzeugt wird.

### config:

Gerätekonfiguration ändern, die Konfigurationsdaten stehen im Textkörper der Mail. Wie das genau geht, ist in Abschnitt 4 erklärt.

### zone:

Die Zeitzone festlegen, z.B. „zone +0100“. Dieser Parameter ist Teil der Konfigurationsdaten und läßt sich darum auch über das „config“-Kommando einstellen, wie weiter unten beschrieben. Diese Vereinfachung ist vorgesehen, weil auch die jährliche Umstellung auf Sommerzeit und zurück über Änderung der Zeitzone geschehen muß.

### update:

Bringt den IO-Mailer in einen Sondermodus fürs Firmware-Update. Sie haben anschließend fünf Minuten Zeit, ein TCP-Socket zu Port 8000 des IO-Mailers zu öffnen und ihm eine Textdatei mit der neuen Firmware zu schicken. Vorsicht: Bis das Update erfolgreich ausgeführt wurde (oder bis die fünf Minuten um sind), steht der IO-Mailer für seine normale Funktion nicht zur Verfügung!

Damit ein Mail-Teilnehmer eines dieser Kommandos verwenden kann, muß er in den meisten Fällen ausdrücklich dazu berechtigt sein. Mehr dazu später, im Abschnitt über Zugriffsbeschränkungen.

## 4 Konfiguration

### 4.1 XML

Der IO-Mailer wird konfiguriert, indem man ihm eine Mail mit dem Betreff „config“ schickt, die als Text ein XML-Dokument mit den Konfigurationsdaten enthält. Konfiguration über ein serielles Terminal ist ebenfalls möglich, die zu verwendenden XML-Dokumente sind aber in beiden Fällen die gleichen.

Die Schnittstellenparameter für das serielle Terminal sind 9600 Baud, 8 Bit, Software-Handshake. Drücken Sie am Terminal einmal die Leertaste, um das Tastaturmenü anzuzeigen, wählen Sie dann den Menüpunkt „configure“ und übertragen Sie die vorbereiteten Konfigurationsdaten mit der Terminal-Funktion „Textdatei senden“.

#### 4.1.1 Wieso XML?

XML (die „extensible markup language“) beschreibt Sprachregeln, mit deren Hilfe sich die Konfigurationsdaten für den IO-Mailer übersichtlich in eine Art Baumstruktur mit benannten Zweigen gliedern lassen. Eine etwas einfachere Syntax, wie z.B. die von .ini-Dateien unter Windows, hätte zwar möglicherweise den gleichen Zweck erfüllt. XML ist aber einfach ein weit verbreiteter Standard, mit dem Vorteil, dass zahlreiche Hilfsmittel erhältlich sind, die den Umgang mit solchen Dokumenten erleichtern. Beispielsweise können Sie eine XML-Datei, wie der IO-Mailer sie als Antwort auf das Mail-Kommando „info“ schickt, mit dem Internet Explorer 5 direkt öffnen und betrachten, und die Baumstruktur Zweig für Zweig durchblättern.

### 4.1.2 Unterschiede zu HTML

Falls Sie bereits mit HTML vertraut sind, wird Ihnen einiges an XML bekannt vorkommen. Es gibt aber auch Unterschiede, wobei die wichtigsten sind:

- Groß-/Kleinschreibung von Element- und Attributnamen wird beachtet.
- Zu jedem öffnenden gehört auch ein schließendes Strukturelement, d.h. `<name>...</name>`. Alleinstehende Elemente sind nur erlaubt, wenn sie durch eine besondere Form der schließenden spitzen Klammer als solche gekennzeichnet sind: `<name/>`.
- Attributwerte müssen stets in Anführungszeichen eingeschlossen sein, also z.B. `<name attr="1"/>`
- Es gibt keine Codierungen für Sonderzeichen, mit Ausnahme der fünf Zeichen `<`, `>`, `&`, `„`, `'` (vgl. untenstehende Tabelle). Beispielsweise `&auml;` anstelle von `'ä'` ist also *nicht* erlaubt.
- Wie bei HTML gibt es auch bei XML für jedes Dokument ein Strukturelementpaar, das den gesamten Dokumentinhalt einschließt, oder anschaulich gesagt: Die Baumstruktur, in die das Dokument gegliedert ist, geht von genau einer Wurzel aus. Aber während diese Wurzel für HTML-Dokumente in jedem Fall `<HTML>` heißt, ist ihr Name bei XML-Dokumenten je nach Anwendung definiert, und für die Konfigurationsdaten des IO-Mailers ist dieser Name `<io-mail.1>`.

### 4.1.3 Arbeiten mit Dokumentvorlagen

Der IO-Mailer kann mit dem Mail-Kommando „template“ ein Rohgerüst des zur Konfiguration benötigten XML-Dokuments erzeugen, in das Sie lediglich noch Ihre eigenen Konfigurationsdaten einzutragen brauchen. Da man den Mailer aber bereits konfiguriert haben müsste, damit er überhaupt ein solches Mail-Kommando verarbeiten kann, steht die gleiche Funktion auch noch einmal am seriellen Terminal

## W&T

zur Verfügung. Wählen Sie hier im Tastermenü den Punkt „full XML template“.

Auch wenn Sie bislang weder XML noch HTML kennen und sich nur über die vielen spitzen Klammern wundern, sollten Sie mit dem Ausfüllen dieser Formulare keine größeren Schwierigkeiten haben, solange Sie die folgenden Hinweise beachten:

### Sonderzeichen.

In den Texten, die Sie selber eingeben, dürfen die spitzen Klammern selber sowie drei weitere Sonderzeichen nicht vorkommen, und müssen durch Codierungen gemäß der folgenden Tabelle ersetzt werden:

Sonderzeichen	Umschreibung
<	&lt;
>	&gt;
"	&quot;
'	&apos;
&	&amp;

In manchen Texten kann es außerdem sinnvoll sein, Steuerzeichen einzufügen: Wagenrücklauf, Zeilenvorschub, Tabulatorsprung. Zu dem Zweck wurden (als Besonderheit des IO-Mailers, kein Bestandteil des XML-Standards) die folgenden Codierungen eingeführt:

Steuerzeichen	Umschreibung
CR	\r
LF	\n
TAB	\t
\	\\

### Leere Felder ausfüllen.

Einfachen Text können Sie zwischen zusammengehörigen öffnenden und schließenden Strukturelementen eintragen, Attributtext zwischen Anführungszeichen. Beispiel:

vorher :

```
<repeat></repeat>
```

## W&T

```
<action var="" val=""/>
```

nachher:

```
<repeat>00:00:01</repeat>  
<action var="Y1" val="0"/>
```

Lange Texte.

Es ist nicht erforderlich, dass öffnendes und schließendes Strukturelement in derselben Zeile stehen, und auch der Text selber darf über mehrere Zeilen verteilt werden. Beispiel:

vorher:

```
<subject></subject>  
<message></message>
```

nachher:

```
<subject>  
ziemlich lange Betreffzeile, jedenfalls für unsere Verhältnisse  
</subject>  
<message>Etwas längerer Absatz.  
Zeilenumbruch,  
    linker Rand  
und Größe der Wortzwischenräume sind  
bei der Eingabe übrigens egal.</message>
```

Löschen und Vervielfältigen.

Auch wenn in der ursprünglichen Dokumentvorlage jedes Feld genau einmal vorkommt: Viele dürfen auch mehrfach verwendet werden, viele können ganz entfallen. Dabei ist das vom IO-Mailer erzeugte Dokument so formatiert, dass man normalerweise nichts falsch macht, solange man immer nur ganze Zeilen löscht oder vervielfältigt. Beispiel (mehr als einen DNS-Server angeben):

vorher:

```
<dns></dns>
```

nachher:

```
<dns>172.16.232.14</dns>
```

```
<dns>172.16.232.15</dns>
```

Noch ein Beispiel, bei dem unbenutzte Elemente und Attribute gelöscht werden:

vorher:

```
<trigger op="DELTA" var="N1" val="3" delay="" recover="" />
```

```
<repeat></repeat>
```

```
<mailto>test@xyz</mailto>
```

nachher:

```
<trigger op="DELTA" var="N1" val="3"/>
```

```
<mailto>test@xyz</mailto>
```

Zwei wichtige Fälle sind nur noch zu erwähnen, wo es mit dem Kopieren bzw. Löschen einzelner Zeilen nicht getan ist. Erstens darf der *alarm*-Zweig mehrere *event*-Zweige enthalten, in der Dokumentvorlage ist es aber zunächst nur einer. Wie hier eine korrekte Konfiguration mit mehreren *event*-Zweigen auszusehen hat, schauen Sie sich am besten einfach in den Beispielen an, weiter unten im Abschnitt „Ereignisse und Aktionen“.

Der andere wichtige Fall hat damit zu tun, dass die Konfiguration in neun unabhängige Hauptzweige unterteilt ist, von denen man gelegentlich einzelne aus dem XML-Dokument entfernen will (mehr dazu auch im nächsten Abschnitt). Jeder Zweig beginnt und endet mit einer Zeile, die das öffnende, bzw. schließende Strukturelement zu seinem Namen enthält, also z.B. `<access>` und `</access>` für den *access*-Zweig. Es ist nun sowohl erlaubt, nur den Inhalt eines Zweiges zu löschen, so dass diese erste und letzte Zeile übrig bleiben, als auch den kompletten Zweig selber, also alle Zeilen von der ersten bis zur letzten einschließlich. Beides hat aber unterschiedliche Auswirkungen: Wird nur der Inhalt eines Hauptzweiges gelöscht, so löscht man damit auch alle zugehörigen Daten aus der Gerätekonfiguration (das will man nur in Ausnahmefällen). Löscht man dagegen den Hauptzweig

selber, bleibt der entsprechende Abschnitt der Gerätekonfiguration unverändert.

Kommentare.

Der Text, der zwischen `<!--` und `-->` steht, wird ignoriert. Sie können damit auch ganze Abschnitte Ihres Dokuments quasi ausblenden. Es ist nur nicht erlaubt, Kommentare ineinander zu verschachteln. Beispiel:

```
<!-- Hmm... Ist das sinnvoll? -->
<!--
<suspect>*</suspect>
-->
```

### 4.1.4 Die Unterteilung in Hauptzweige

Die Konfiguration ist in acht Hauptzweige unterteilt, die in den folgenden Abschnitten beschrieben werden. Diese Hauptzweige können unabhängig voneinander konfiguriert werden. Das bedeutet, dass man z.B. die Zeitzone des Mailers ändern kann, ohne auch die Triggerbedingungen neu definieren zu müssen. Für jeden Zweig, der beim Verschieken von Konfigurationsdaten überhaupt erwähnt wird, werden aber alle vorhandenen Daten gelöscht, bevor neue übernommen werden. Wer also z.B. einen neuen DNS-Server eintragen will, muß auch die Time- und Mail-Server wissen und erneut angeben. Ein prinzipielles Hindernis für Änderungen sollte das aber nicht sein, da die aktuellen Konfigurationsdaten jederzeit aus dem IO-Mailer selber abrufbar sind (Mail-Kommandos „info“ und „template“).

Überhaupt ist die empfohlene Methode für geringfügige Änderungen, auch wirklich nur den oder die betroffenen Zweige zu ändern. In dem o.g. Beispiel mit dem zu ändernden DNS-Server würde man also eine Mail mit dem Betreff „info servers“ schicken, das auf die Weise erhaltene XML-Dokument editieren und dann mit dem Betreff „config“ zurückschicken. Man vermeidet damit zwei mögliche Probleme: Erstens stört es unnötig den Betrieb des IO-Mailers, wenn be-



## W&T

stimmte Zweige neu geschrieben werden, obwohl sich dabei inhaltlich gar nichts an ihnen ändert (*report*: die Meßwertreihen werden gelöscht, *alarm*: die Ereignisverarbeitung wird neu initialisiert). Und außerdem gibt es Zweige, die einige Mail-Teilnehmer sowieso nur anzeigen, aber nicht ändern dürfen (*servers*, *pop-account*, *access*).

Von dieser Unterteilung in unabhängige Zweige wird auch bei den folgenden Beispielen sehr ausgiebig Gebrauch gemacht. Im allgemeinen sind das nämlich komplette XML-Dokumente, die in dieser Form auch direkt vom IO-Mailer akzeptiert würden, die als Besonderheit aber jeweils nur einen einzigen Hauptzweig enthalten. In der Tat ist kein einziges Beispiel dabei, das mehrere Hauptzweige gleichzeitig ändert. Nur um Mißverständnissen vorzubeugen, soll hier darum einmal angedeutet werden, wie man mehrere dieser Beispiele kombinieren würde, um mehrere Zweige gleichzeitig zu ändern. Das hier wäre die *falsche* Methode:

```
<io-mail.1>
  <servers>
  ...
</servers>
</io-mail.1>    <!-- So bitte nicht! Alles nach dem -->
<io-mail.1>    <!-- ersten </io-mail.1> wird ignoriert! -->
  <pop-account>
  ...
</pop-account>
</io-mail.1>
```

Und das hier die richtige:

```
<io-mail.1>
  <servers>
  ...
</servers>
  <pop-account>
  ...
</pop-account>
</io-mail.1>
```

Oder anders gesagt: Das Strukturelementpaar, das das gesamte Dokument einschließt (in unserem Fall ist das `<io-mail.1>...</io-mail.1>`) darf naturgemäß nur einmal vorkommen. Andernfalls hätte man es mit mehreren Konfigurationsdateien zu tun, aber Sie dürfen dem IO-Mailer immer nur eine einzige Datei auf einmal schicken.

### 4.1.5 Zeitangaben

An mehreren Stellen innerhalb der Konfigurationsdaten kommen Uhrzeiten und Zeitspannen vor. Das hierbei zu verwendende Format ist in allen Fällen (mit Ausnahme der Zeitzone) HH:MM:SS, also z.B. 04:30:00 für halb fünf Uhr früh und 00:00:10 für zehn Sekunden. Zeitspannen können alternativ auch einfach als eine Anzahl Sekunden angegeben werden, z.B. 90 für eineinhalb Minuten oder 0.25 für eine viertel Sekunde.

### 4.2 Netzwerkumgebung (servers)

In Ihrem Netzwerk sollten folgende Server vorhanden sein:

- ein DNS-Server. Nicht unbedingt erforderlich, aber andernfalls müssen sie alle anderen Server direkt über deren IP-Adresse statt über einen Hostnamen angeben.
- ein SMTP-Server und ein POP3-Server zum Senden bzw. Empfangen von Mail. Diese Dienste sind häufig zusammen auf ein und demselben Server installiert (und im Falle von Microsoft Exchange Server z. B. ist es sogar ein und dasselbe Produkt).
- ein Timeserver nach RFC-868. Dringend empfehlenswert, damit der IO-Mailer korrekte Zeitstempel erzeugen kann. Es gibt weltweit erreichbare Timeserver im Internet (z.B. [clock.psu.edu](http://clock.psu.edu), [swisstime.ethz.ch](http://swisstime.ethz.ch)) sowie kostenlose Software, mit der Windows-Rechner als Timeserver eingerichtet werden können, mehr dazu im Anhang. Am einfachsten ist es aber, wenn Sie einen eigenen Unix- oder Linux-Server zur Verfügung haben. Dort brauchen Sie lediglich sicherzustellen, dass in der Datei „inetd.conf“

der als „time“ bezeichnete Dienst auf TCP-Port 37 aktiviert ist.

Beispiel:

```
<io-mail.1>
  <servers>
    <dns>172.16.232.15</dns>
    <dns>172.16.232.14</dns>
    <smtp>mail.firma.de</smtp>
    <pop3>mail.firma.de</pop3>
    <time>atropos.firma.de</time>
    <time>clock.psu.edu</time>
    <time>time.iien.it</time>
    <connect>5.00</connect>
    <terminate>10.00</terminate>
    <response>60.00</response>
  </servers>
</io-mail.1>
```

Zu jedem Server-Typ können mehrere alternative Adressen, bzw. Hostnamen angegeben werden. Dabei versucht der IO-Mailer im Normalfall, sich mit dem jeweils zuerst aufgeführten zu verbinden, und erst wenn das fehlschlägt, geht er der Reihe nach die übrigen Adressen durch.

Mehr als einen DNS- bzw. Time-Server anzugeben kann auf keinen Fall schaden. Eine Konfiguration mit mehreren SMTP-Servern ist schon eher unüblich, und mehrere POP3-Server zur Auswahl sind normalerweise nicht sinnvoll (aufgrund der Funktionsweise dieses Dienstes: Der POP3-Server repräsentiert eine Sammlung von Postfächern, und wenn er ausfällt, steht deren Inhalt einfach nicht mehr zur Verfügung).

Der Connect-Wert, den Sie hier eintragen, bestimmt, wieviel Zeit der IO-Mailer seinen Servern zum Akzeptieren einer TCP-Verbindung einräumt. Solange nichts anderes angegeben ist, gelten hier als Standardwert 5 Sekunden. Noch größere Werte sind normalerweise unpraktisch, weil es dann sehr lange dauert, bis der evtl. Ausfall eines Servers erkannt wird. Noch schlimmer sind dagegen zu kleine Werte, weil

dann einfach keine Netzwerkverbindungen mehr zustande kommen.

Innerhalb der diversen TCP-Verbindungen führt der IO-Mailer mit seinen Servern längere Dialoge, wobei es leider normal ist, dass stark ausgelastete Server für die einzelnen Antworten sehr viel Zeit brauchen. Mit dem Response-Wert wird eine Obergrenze für diese Antwortzeiten festgelegt. Wird die angegebene Zeit überschritten, beendet der IO-Mailer die Verbindung. Als Standardwert gilt hier 1 Minute.

Das Gegenstück zum Connect-Wert ist der Terminate-Wert, in dem eingetragen wird, binnen welcher Zeit Server einen Verbindungsabbau quittieren müssen. Als Standardwert gelten 10 Sekunden. TCP-Verbindungen werden normalerweise erheblich schneller getrennt, innerhalb weniger Millisekunden.

### 4.3 Postfachzugang (pop-account)

Diese Informationen benötigt der IO-Mailer zum Empfang von Mail. Beispiel:

```
<io-mail.1>
  <pop-account>
    <mailbox>eabox1</mailbox>
    <password>caligula</password>
    <interval>00:01:00</interval>
  </pop-account>
</io-mail.1>
```

Vorsicht: Das Postfach, das Sie hier angeben, muß dem Mailer alleine gehören! Alle Mails, die er gelesen hat, löscht der IO-Mailer nämlich sofort, auch solche, die er inhaltlich gar nicht versteht. Versuchen Sie darum niemals, den Mailer das Postfach eines Menschen mitbenutzen zu lassen! Und auch mehrere Mailer ein Postfach teilen zu lassen, wäre nicht sinnvoll. (Mails an dieses Postfach würden dann nur jeweils einen der Mailer erreichen, wobei nicht vorhersagbar wäre, welchen.)

Beachten Sie außerdem, dass der IO-Mailer keine Passwort-verschlüsselung, auch bekannt als MD5- oder Challenge-Response-Authentifizierung, unterstützt. Das bedeutet, dass das Kennwort bei jeder Abfrage der Mailbox unverschlüsselt übers Netzwerk zum POP3-Server gesendet wird, und für andere Netzteilnehmer mit relativ geringem technischen Aufwand abzufangen ist. Abgesehen davon nimmt das Passwort auch keine Sonderrolle innerhalb der Konfigurationsdaten ein und läßt sich von den dazu berechtigten Personen zusammen mit dem Rest der Konfiguration einfach aus dem Gerät abfragen.

Das eigentliche Sicherheitsrisiko liegt dabei weniger in der Möglichkeit, dass jemand unberechtigt Zugriff auf das Postfach des IO-Mailers erhält, denn die Mails, die dort liegen, stellen normalerweise keine großen Geheimnisse dar. Sie sollten vielmehr darauf achten, dass aus der Kenntnis des Passworts möglichst keine weiteren Rechte erlangt werden können. Insbesondere sollte man hier nicht das Passwort eines menschlichen Benutzers „wiederverwenden“.

Das Abfrageintervall schließlich bestimmt, wie schnell der IO-Mailer auf eingehende Mail-Kommandos reagieren kann. Hier sollte man, um unnötige Netzwerkbelastung zu vermeiden, nicht zu ehrgeizig sein. Eine Reaktionszeit in der Größenordnung von einer Minute reicht im allgemeinen aus.

#### 4.4 Absenderadresse des Mailers (device-id)

Beispiel:

```
<io-mail.1>
  <device-id>
    <host>eabox1.firma.de</host>
    <address>eabox1@firma.de</address>
    <name>E/A-Server im Treppenhaus</name>
    <owner>f.mustermann@firma.de</owner> <!-- optional -->
  </device-id>
</io-mail.1>
```

Diese Informationen verwendet der IO-Mailer beim Senden von Mail. Den eigenen Hostnamen gibt er an, um sich gegenüber dem SMTP-Server zu identifizieren. Die Mail-Adresse wird als Absender auf allen erzeugten Mails vermerkt und ermöglicht es dem Empfänger, dem IO-Mailer zu antworten (hilfreich vor allem für das Abbestellen unerwünschter Benachrichtigungen). Der „Name“ des Mailers schließlich ist ein beliebiger Text, der beim Empfänger anstelle von oder zusammen mit der Absenderadresse angezeigt wird. Hier könnten Sie einen kurzen Hinweis auf Funktion, Standort oder Betreiber des Mailers eingeben.

Das Feld <owner> sollte die Mail-Adresse einer Person enthalten, die für den Betrieb des IO-Mailers verantwortlich ist. Erstens wird diese Adresse in den Kopfzeilen aller automatisch erzeugten Mails vermerkt, wo jemand, der bei Problemen einen Ansprechpartner sucht, sie wahrscheinlich auch finden wird. Außerdem schickt der IO-Mailer hierhin Benachrichtigungen über diverse besondere Vorkommnisse, z. B. wenn ein Empfänger sich mit dem Mail-Kommando „unsubscribe“ selber aus der Konfiguration gelöscht hat. In vielen Fällen ist diese Angabe aber überflüssig, und zwar dann, wenn Sie den IO-Mailer stets per Mail konfigurieren (also nicht über die serielle Schnittstelle). In dem Fall wird als Ansprechpartner automatisch derjenige eingetragen, der zuletzt den Konfigurationszweig <alarm> geändert hat.

### 4.5 Erscheinungsbild des Gerätestatus (report)

Jede Mail, die aufgrund eines Ereignisses (oder in Antwort auf das Kommando „status“) versandt wird, enthält als Text mindestens eine Auflistung der sieben wichtigsten Gerätevariablen, was im einfachsten Fall z.B. so aussehen kann:

Thu, 29 Jun 2000 13:52:44

X0 = 0

X1 = 1

Y0 = 0

Y1 = 0

N0 = 37

## W&T

N1 = 4

DN = -33

Da das für den Empfänger der Mail ziemlich nichtssagend ist, kann jeder Variable ein Kommentartext zugeordnet werden, den Binärvariablen (das sind X0, X1, Y0, Y1, und noch ein paar andere, siehe auch Tabelle weiter unten) außerdem je ein Text für den Zustand 0 und 1.

Abgesehen davon läßt sich über diesen Abschnitt der Konfiguration auch noch die Aufnahme von Meßreihen steuern. Und zwar können für die beiden Zähler N0, N1 bis zu 256 zurückliegende Werte gespeichert werden, in einem frei definierbaren Zeitraster.

Hier ein komplettes Beispiel:

```
<io-mail.1>
  <report>
    <label var="X0" alias="Lichtschranke"
      state0="frei" statel="unterbrochen"/>
    <label var="X1" alias="Klingeltaster"
      state0="losgelassen" statel="betätigt"/>
    <label var="Y0" alias="Beleuchtung" state0="aus" statel="ein"/>
    <label var="N0" alias="Besucher"/>
    <history var="N0" depth="4" step="00:00:15"/>
    <history var="N1" depth="6" step="00:00:10"/>
  </report>
</io-mail.1>
```

Und eine mögliche Ausgabe, die dadurch erzeugt würde:

Wed, 2 Aug 2000 17:02:04

X0 = 0 (Lichtschranke frei)

X1 = 0 (Klingeltaster losgelassen)

Y0 = 1 (Beleuchtung ein)

Y1 = 0

N0 = 6 (Besucher)

N1 = 10

17:02:00, N0 = 5 (+1)

## W&T

17:01:45, N0 = 2 (+3)

17:01:30, N0 = 0 (+2)

17:01:15, N0 = 0

17:02:00, N1 = 9 (+1)

17:01:50, N1 = 6 (+3)

17:01:40, N1 = 6

17:01:30, N1 = 0 (+6)

17:01:20, N1 = 0

17:01:10, N1 = 0

Die Zahlen in Klammern hinter den einzelnen Meßreihenwerten geben dabei jeweils die Differenz zum nächstjüngeren Wert an. Im Falle der ersten Zeile jeder Liste ist das die Differenz zu dem aktuellen Wert, der nicht mehr in der Liste selber, sondern in der Zusammenfassung der Variablen am Anfang der Mail steht.

Eine weitere Besonderheit bei der jüngsten Differenz in der Meßreihe ist noch, dass sie im allgemeinen für einen kürzeren Zeitraum gilt als alle anderen Werte. Wie kurz genau, hängt vom Zeitpunkt der Abfrage ab und ist damit relativ zufällig. Wer sich daran stört, weil dieser Wert sich dadurch schlechter mit den anderen vergleichen läßt, kann über den zusätzlichen Parameter *grid* veranlassen, dass bei der Aufzeichnung der Meßreihen ein feineres Raster verwendet wird als bei ihrer Ausgabe. Was das bewirkt, zeigt das folgende Beispiel. Dabei gilt für beide Spalten *depth*="4" und *step*="01:00:00", für die rechte Spalte außerdem *grid*="12". (Für die linke wurde kein *grid*-Wert angegeben, was *grid*="1" entspricht.) Außerdem nehmen wir an, dass die Abfrage um 17:04:25 stattfindet, und der aktuelle Wert für N0 2135 ist.

17:00:00, N0 = 2108 (+19)            16:05:00, N0 = 1988 (+147)

16:00:00, N0 = 1956 (+152)        15:05:00, N0 = 1856 (+132)

15:00:00, N0 = 1818 (+138)        14:05:00, N0 = 1703 (+153)

14:00:00, N0 = 1674 (+144)        13:05:00, N0 = 1558 (+145)

Zu beachten ist noch, dass sich bei Angabe eines *grid*-Parameters die nutzbare Länge der Meßreihe verkürzt: Die Werte *depth* und *grid* miteinander multipliziert dürfen höchstens



256 ergeben. In dem gezeigten Beispiel mit `grid="12"` wäre also `depth="21"` der größte sinnvolle Wert.

### 4.6 Zeitzone (zone)

Gibt die Zeitzone an, in der der IO-Mailer steht, z.B. +0100 für MEZ, +0200 für MESZ (Mittleuropäische Zeit bzw. Sommerzeit). Wie durch diese Beispiele bereits angedeutet, werden hier die Stunden und Minuten Verschiebung gegenüber der Weltzeit (GMT, früher auch als Greenwich-Zeit bezeichnet) angegeben, jeweils zweistellig, ohne Trennzeichen dazwischen. Beispiel:

```
<io-mail.1>  
  <zone>+0100</zone>  
</io-mail.1>
```

Wichtig: Die Umschaltung zwischen Sommer- und Winterzeit erfolgt nicht automatisch, sondern erfordert, dass der Benutzer tätig wird und die eingestellte Zeitzone ändert.

Der IO-Mailer verwendet die Zonen-Information, um die Weltzeit, wie er sie von seinem Timeserver erhält, in Ortszeit umzurechnen. Angabe einer falschen Zeitzone bewirkt darum, dass die Uhrzeiten, die in die Statusberichte des Mailers eingefügt werden, ebenfalls falsch, nämlich gegenüber der tatsächlichen Ortszeit verschoben sind.

Vorausgesetzt, die vom Timeserver empfangene Weltzeit ist korrekt, läßt sich aber auch bei falscher Zoneninformation noch feststellen, wann eine Mail wirklich gesendet wurde. In den Kopfzeilen der Mail ist nämlich sowohl die vermeintliche Ortszeit selber vermerkt, als auch die Zeitzone, für die diese Ortszeit gelten soll. In der Tat werten manche Mail-Clients diese Informationen bereits aus, um den Zeitpunkt, zu dem eine Mail gesendet wurde, sofort auf die Ortszeit des Empfängers umzurechnen.

## 4.7 Zugriffsbeschränkungen (access)

Bevor der IO-Mailer ein Mail-Kommando ausführt, prüft er, ob der Absender der Mail zu der betreffenden Aktion berechtigt ist. Dazu lassen sich zwei bevorrechtigte Personengruppen einrichten: Benutzer (*user*) und Administratoren (*admin*). Vereinfacht gesagt werden mindestens Benutzerrechte benötigt, um irgend etwas am IO-Mailer zu verändern, und man braucht Administratorrechte, um kritische Details der Konfiguration zu ändern. (Als „kritisch“ gelten dabei solche Änderungen, mit denen man sich selber ausperren kann, die sich also möglicherweise nicht mehr per Mail, sondern nur noch vor Ort am seriellen Terminal rückgängig machen lassen.) Eine dritte bevorrechtigte Personengruppe sind noch die vertrauenswürdigen Empfänger, zu der aber von vornherein jeder gehört, den man nicht ausdrücklich als *suspect* ausschließt. Im Detail sieht das so aus:

Berechtigungsstufe	Mail-Kommandos
jedermann	help
vertrauenswürdiger Empfänger	status unsubscribe resume
Benutzer	set zone info template config (Zweige zone, schedule, serial, device-id, report, alarm)
Administrator	update config (Zweige access, servers, pop-account)

Eine sinnvolle Politik könnte z.B. so aussehen, dass allen Mitarbeitern der eigenen Firma Benutzerrechte eingeräumt werden, und nur einigen ausdrücklich genannten darüber hinaus auch Administratorrechte, wie in der folgenden Konfiguration:

```
<io-mail.1>
```

```
<access>
  <suspect>*/</suspect>
  <user>firma.de/</user>
  <admin>f.mustermann@firma.de</admin>
  <admin>m.dietrich@firma.de</admin>
</access>
</io-mail.1>
```

Einträge, die ein „@“ enthalten, stellen dabei einfache Mail-Adressen dar, Einträge ohne „@“ gelten für alle Absender aus einer kompletten Mail-Domain, und „\*“ ist ein Joker, der auf jede beliebige Adresse paßt. Von mehreren passenden Einträgen gilt im Zweifelsfall der, der die speziellste Beschreibung darstellt.

Durch den Eintrag „firma.de“ würden hier also z.B. „frank@firma.de“ und „klaus@firma.de“ Benutzerrechte eingeräumt, ebenso „thomas@zweigstelle.firma.de“, nicht aber „stefan@andere-firma.de“ und auch nicht „martin@firma.definitiv.de“. Der Eintrag „\*“ unter *suspect* weist allen Mail-Teilnehmer außerhalb der Firma die niedrigste Berechtigungsstufe zu, wobei die wichtigste Auswirkung ist, dass sie nicht mehr das „status“-Kommando verwenden können. Das kann sinnvoll sein, wenn der Mailer über sensible Informationen verfügt, die nicht nach außen gelangen sollen, z.B. ob bestimmte Türen im Gebäude offenstehen.

Eine Schwachstelle bei all diesen schönen Überlegungen ist nur, dass sich Mail-Absenderadressen fälschen lassen. Wer so etwas tut, muß zwar einige Probleme überwinden, von denen nicht das geringste ist, dass ihn keine Antworten vom IO-Mailer mehr erreichen werden. (Antworten werden stets an dieselbe Adresse geschickt, die auch zur Prüfung der Berechtigung verwendet wurde, d.h. Reply-To: Kopfzeilen werden nicht beachtet.) An der grundsätzlichen Möglichkeit eines solchen Angriffs ändert das aber nichts. Wirkliche Sicherheit dagegen bietet nur eine radikale Lösung:

```
<io-mail.1>
  <!-- keine bevorrechtigten Zugriffe per Mail mehr erlauben -->
  <access>
```

```
<suspect>*</suspect>  
</access>  
</io-mail.1>
```

In dieser Konfiguration ist „help“ das einzige Kommando, das noch per Mail ausgeführt werden kann, und Konfigurationsänderungen lassen sich nur noch über den seriellen Port direkt am Gerät vornehmen. Man könnte sogar noch einen Schritt weitergehen und in den Konfigurationsdaten einfach die Angaben für POP3-Server und Postfach weglassen, so dass der IO-Mailer nur noch senden, aber nicht mehr empfangen kann. Auf die Weise würde man aber auch die Möglichkeit einbüßen, sich durch ein help-Kommando auf unkomplizierte Weise von der Funktionsfähigkeit des Mailers zu überzeugen.

## 4.8 Zeitabhängige Steuerung (schedule)

Die Aktionen des IO-Mailers lassen sich in begrenztem Umfang auch in Abhängigkeit von Tageszeit und Wochentag steuern. Dazu dienen zwei zusätzliche binäre Variablen WE und NT, die den Zustand „Wochenende“ bzw. „Nacht“ anzeigen. Um sie zu benutzen, müssen Sie zunächst festlegen, welche Zeiträume als Nacht und Wochenende gelten sollen, z.B.:

```
<io-mail.1>  
  <schedule>  
    <night from="22:00:00" to="06:00:00"/>  
    <weekend from="Fr 22:00:00" to="Mo 06:00:00"/>  
  </schedule>  
</io-mail.1>
```

Anmerkungen:

- Die Angabe des Wochentages kann mit deutschen oder englischen Abkürzungen erfolgen, wobei außerdem nur die ersten zwei Buchstaben beachtet werden. So wären

z.B. „Di“, „Die“, „Tu“ und „Tue“ allesamt gültige Bezeichnungen für Dienstag.

- Solange die Echtzeituhr des IO-Mailers nicht gestellt werden konnte (z.B. weil keiner der konfigurierten Time-server erreichbar war, oder weil überhaupt keiner angegeben ist), ist die zeitabhängige Steuerung unwirksam, und die Variablen WE und NT sind immer 0.
- Die zeitabhängigen Variablen werden auch im Gerätestatus angezeigt, allerdings nur wenn sie überhaupt benutzt werden, d.h. wenn (wie hier gerade gezeigt) Zeiten für ihre Funktion festgelegt sind. Für das Erscheinungsbild des Gerätestatus würden sich dann zwei zusätzliche Zeilen im <report>-Zweig anbieten, die ungefähr so aussehen könnten:

```
<label var="WE" state0="Werktag" state1="Wochenende"/>
<label var="NT" state1="Nacht"/>
```

### 4.9 Die serielle Schnittstelle (serial)

An den IO-Mailer kann entweder ein serielles Terminal für Konfigurations- und Diagnosezwecke angeschlossen sein, oder ein anderes Peripheriegerät mit serieller Schnittstelle, das dann auf diese Weise Mails versenden oder per Mail ferngesteuert werden kann. Entsprechend diesen zwei Fällen kann die serielle Schnittstelle des Mailers sich entweder im *Terminalmodus* oder im *Peripheriemodus* befinden, wobei es für den Peripheriemodus zusätzliche Konfigurationsdaten gibt:

```
<io-mail.1>
  <serial>
    <line baud="9600" data="8" stop="1" parity="N" flow="X"/>
    <escchar>27</escchar>
    <packet mode="D" timeout="0:00:10" chopmsg="[...]" />
    <dpacket endchar="13" filter="1" />
  </serial>
</io-mail.1>
```

Mit *line* werden die seriellen Schnittstellenparameter festgelegt. Verfügbare Baudraten sind 1200, 2400, 4800, 7200, 9600, 14400, 19200, 28800, 38400, 57600. Datenbits können 7 oder 8 sein, Stopbits 1 oder 2. N, E und O stehen für keine, gerade und ungerade Parität (none, even, odd). Die Flußkontrolle (*flow*) kann entweder N für kein Handshake, oder X für Software-Handshake (XOn/XOff) sein. Hardware-Handshake wird nicht unterstützt, da die serielle Schnittstelle des Mailers keine Handshake-Leitungen hat. Denken Sie bitte daran: Die Schnittstellenparameter im Terminalmodus sind, unabhängig davon, was hier konfiguriert ist, stets 9600 Baud, 8N1, Software-Handshake.

Der IO-Mailer startet beim Einschalten standardmäßig im Peripheriemodus. Um stattdessen den Terminalmodus zu aktivieren, gibt man, wie oben bei der Inbetriebnahme beschrieben, während des Einschaltens wiederholt „x“ ein, bis die Aufforderung zur Eingabe der IP-Adresse erscheint, und überspringt dann diese Abfrage, mit Enter oder Escape. Wer das zu umständlich findet, kann mit *escchar* ein Zeichen definieren, bei dessen Eingabe der Mailer vom Peripheriemodus in den Terminalmodus zurückkehrt. Angegeben wird hier ein ASCII-Code, z.B. 27 für Escape. Bedenken sie aber, dass es schwer vorhersagbare und vermutlich unangenehme Folgen hat, wenn das Peripheriegerät selber einmal, aus was für Gründen auch immer, dieses Sonderzeichen senden sollte.

Serielle Daten, die Ereignisse auslösen sollen, werden vom Mailer zunächst in Pakete unterteilt. Auf welche Weise, wird über die *packet*- und evtl. *dpacket*-Parameter festgelegt. Es gibt zwei Hauptbetriebsarten: Kommandomodus (*mode="C"*), wobei das Peripheriegerät Empfänger und Inhalt jeder zu sendenden Mail selbst bestimmen kann, und den Datenmodus (*mode="D"*), wobei die Ausgaben des Gerätes analysiert werden, ohne dass sie auf besondere Weise für die Zusammenarbeit mit dem IO-Mailer angepaßt sein müßten. Ein Paket im *Kommandomodus* hat das folgende Format:

Betreff EOL

Empfängeradresse(n) EOL

## W&T

```
Textzeile EOL
...
Textzeile EOL
EOL
```

Werden mehrere Empfängeradressen angegeben, dient Komma oder Semikolon als Trennzeichen, z.B. „ab@beispiel.de, xy@beispiel.de“ (ohne die Anführungszeichen). Für EOL (end of line) kann ein beliebiges der üblichen Zeilenendezeichen verwendet werden: CR, LF, und die Kombination CR/LF. Das Paketende wird an zwei direkt aufeinanderfolgenden Zeilenenden erkannt.

Serielle Pakete können auch dann als beendet gelten, wenn einfach längere Zeit keine seriellen Daten mehr eintreffen. Wenn nichts anderes angegeben ist, gilt hierfür eine Zeitschranke von einer Minute. Geben Sie *timeout*="0" an, um die Verwendung von Zeitschranken zu unterbinden. Wie bereits erwähnt, endet im Kommandomodus ein Paket außerdem dann, wenn zwei Zeilenenden hintereinander eintreffen. Im Datenmodus kann stattdessen ein spezielles Paketendezeichen definiert werden. Hier könnte man z.B. 10 für LF oder 13 für CR angeben, wenn jede Zeile ein Paket darstellen soll. Mit *filter*="1" kann verhindert werden, dass das Paketendezeichen mit im Puffer abgelegt wird. Das macht vor allem dann einen Unterschied, wenn die seriellen Daten (über den Platzhalter %s) in eine Mail eingefügt werden sollen.

Für die seriellen Daten steht ein 8 KByte großer Empfangspuffer zur Verfügung, Pufferüberlauf ist deswegen relativ unwahrscheinlich. Sollte das dennoch einmal vorkommen, wird an das betroffene Paket an der Stelle, wo es abgeschnitten werden mußte, der über *chopmsg* angegebene String angefügt.

### 4.10 Ereignisse und Aktionen (alarm)

Über diesen Teil der Konfiguration wird die eigentliche Nutzung des IO-Mailers gesteuert. Das Grundprinzip ist

dabei, dass Änderungen an den Eingängen Änderungen an den Ausgängen und/oder das Senden einer Mail bewirken können. Genauer betrachtet haben wir es mit Variablen, Operatoren, Triggerbedingungen, Ereignissen, Aktionen und Empfängerlisten zu tun. Diese Begriffe sollen im folgenden kurz erklärt werden.

## Variablen

Binärvariablen können die Werte 0 und 1 annehmen. Die Zähler sind 32bit-Variablen mit einem Wertebereich von 0 bis 4 294 967 295, der Differenzwert wird als vorzeichen-behaftete Zahl mit einem Wertebereich von -2 147 483 648 bis 2 147 483 647 berechnet. Eine Zuweisung an die Variable TX bewirkt, dass der betreffende String über die serielle Schnittstelle gesendet wird, auslesen lässt sie sich nicht.

Variable	Typ	Bedeutung
X0, X1	binär	Eingänge
Y0, Y1	binär	Ausgänge
TX	Zeichenkette	serieller Sendekanal (Leitung TxD)
N0, N1	Zahlenwert	Eingangszähler
DN	Zahlenwert	Differenz der Zähler, N1-N0
NT, WE	binär	Test auf Nacht bzw. Wochenende
MO, M1	binär	frei verwendbare Speicher ("Merker")

Zur Funktionsweise der Zählervariablen (N0, N1) ist noch anzumerken, dass nicht Flanken, sondern Zyklen gezählt werden, oder anschaulich gesagt: Betätigen und wieder Loslassen eines Tasters an einem Eingang erhöht den zugehörigen Zähler nicht um zwei, sondern um eins.

Die genaue Regel lautet, dass jedesmal gezählt wird, wenn ein Eingang zu dem Zustand zurückkehrt, den er beim Einschalten des Gerätes hatte. Bei dem Taster aus dem Beispiel würde sich der Zählerstand also jedesmal erst beim Loslassen ändern, und zwar unabhängig davon, ob der Taster einen Kontakt öffnet oder schließt. Ob nun wirklich bei steigender oder bei fallender Flanke gezählt wird, wird in vielen Anwendungsfällen aber sowieso keinen großen Unterschied machen.



## Operatoren

Es gibt sechs einfache Operatoren, die jeweils eine Variable gegen einen konstanten Wert vergleichen, sowie als siebten noch den „DELTA“-Operator, dessen Testbedingung als erfüllt gilt, sobald sich der zugehörige Variablenwert um einen Mindestbetrag geändert hat. (Zu diesem Zweck merkt sich der Operator einen Momentanwert seiner Variablen als Referenz. Wann genau er diese Referenzwerte aufnimmt, ist weiter unten erklärt, bei der Beschreibung der Ereignisse.)

Operator	Test auf...	Anmerkung
EQ	Gleichheit	"equal"
NE	Ungleichheit	"not equal"
GT	größer als	"greater than"
LT	kleiner als	"less than"
GE	größer oder gleich	"greater or equal"
LE	kleiner oder gleich	"less or equal"
DELTA	Änderung um einen Mindestbetrag	

## Triggerbedingungen

Eine Triggerbedingung faßt im wesentlichen einen Operator mit seinen zugehörigen Vergleichsobjekten „Variable“ und „konstante Zahl“ zusammen.

Darüber hinaus kann bei Bedarf noch eine Ansprechverzögerung angegeben werden (*delay*). Sie bewirkt, dass das Testergebnis des Operators erst eine Zeit lang ununterbrochen erfüllt sein muß, bevor auch die Triggerbedingung selbst erfüllt ist. Ebenso läßt sich noch eine Erholzeit angeben (*recover*), die umgekehrt bewirkt, dass eine Triggerbedingung eine Zeit lang weiter erfüllt bleibt, auch wenn das Testergebnis des Operators vorübergehend nicht mehr erfüllt ist. (Der Sinn davon ist, zu verhindern, dass etwas als mehrere Ereignisse hintereinander registriert wird, was eigentlich nur ein einziges darstellt.) Sowohl Ansprechverzögerung als auch Erholzeit könnte man als Maßnahmen auffassen, um vorübergehende Störungen der Eingangssignale, z.B. durch prellende Kontakte, zu unterdrücken. Allerdings werden in der Praxis viele Störsignale auch bereits dadurch ausgefiltert, dass die Eingangsvariablen des IO-

Mailers nur in einem Zeitraster von 10 ms aktualisiert werden.

Beispiel: Angenommen, an Eingang X0 ist ein Taster angeschlossen, dann ist die erste der folgenden Triggerbedingungen erfüllt, nachdem der Taster zehnmals betätigt wurde, die zweite, wenn man ihn mindestens eine halbe Sekunde lang gedrückt hält:

```
<trigger op="DELTA" var="N0" val="10"/>  
<trigger op="EQ" var="X0" val="1" delay="0.5"/>
```

### Serielle Triggerbedingungen

Eine serielle Triggerbedingung ist in dem Augenblick erfüllt, da ein Datenpaket eintrifft, und verfällt anschließend sofort wieder. Außerdem wird man im allgemeinen noch verlangen, dass das Paket einen bestimmten Text enthält, wobei die Textsuche wahlweise mit (*case="1"*) oder ohne Beachtung von Groß-/Kleinschreibung (*case="0"*) durchgeführt werden kann. Von den folgenden Beispielbedingungen würde die erste also durch den Text „Ende“ erfüllt, nicht aber durch „Legende“, und die zweite sowohl durch „Alarm“ als auch durch „Feueralarm“.

```
<dtrigger substring="Ende" case="1"/>  
<dtrigger substring="alarm" case="0"/>
```

### Aktionen

Eine „Aktion“ im Sinne des IO-Mailers ist eine Wertzuweisung an eine seiner Variablen. (Das Versenden einer Mail fällt also nicht unter diesen Begriff.) Am üblichsten dürfte dabei das Setzen von Ausgangsvariablen und Merkern sein (Y0, Y1, M0, M1), sowie das Senden von seriellen Daten (Zuweisung an TX), aber wer unbedingt will, kann auch die Zählerstände ändern (N0, N1). Nicht erlaubt dagegen sind Zuweisungen an die Zählerdifferenz, die Tageszeitvariablen und die Eingänge.

### Empfängerlisten

Empfängeradressen können einem bestimmten Ereignis zugeordnet sein oder auch global gelten. Eine Mail wird ver-

sandt, wenn ein Ereignis eintritt, zu dem mindestens eine Empfängeradresse angegeben ist. Falls es global gültige Adressen gibt, löst dementsprechend jedes Ereignis eine Mail aus.

### Ereignisse

Ein Ereignis besteht aus Triggerbedingungen, Aktionen und einer Liste von Mail-Empfängern. Bei mehreren Triggerbedingungen müssen alle gleichzeitig erfüllt sein, um das Ereignis auszulösen (implizite UND-Verknüpfung).

Sobald ein Ereignis eintritt, werden die zugehörigen *Aktionen* unverzüglich ausgeführt (mit einer Reaktionszeit in der Größenordnung von 10 ms), evtl. an den Triggerbedingungen beteiligte DELTA-Operatoren werden zurückgesetzt (indem sie angewiesen werden, einen neuen Referenzwert aufzunehmen), und es wird eine Mail-Anforderung vermerkt, die erst wieder gelöscht wird, wenn tatsächlich eine entsprechende Mail erzeugt werden konnte. Auch wenn der Mail-Versand sich aus verschiedenen Gründen erheblich verzögern kann, ist also sichergestellt, dass keine Mails vergessen werden, nur weil etwa das zugehörige Ereignis (z.B. Betätigung eines Tasters) zu schnell wieder beendet wäre. Mail-Anforderungen sammeln sich aber nicht an, d.h. wenn ein Ereignis mehrmals eintritt, bevor sich Gelegenheit zum Senden bietet, wird trotzdem nur eine einzige Mail erzeugt.

All dies geschieht wie gesagt nur, wenn ein Ereignis gerade neu eintritt, läßt sich aber auch mit Hilfe eines zusätzlichen *repeat*-Parameters in regelmäßigen Abständen wiederholen, solange das Ereignis andauert (d.h. solange alle seine Triggerbedingungen erfüllt sind). Beachten Sie dabei nur, dass Ereignisse, die einen DELTA-Operator enthalten, offensichtlich niemals andauern können und sich darum auch nicht wiederholen lassen.

## 4.10.1 Einfache E/A-Beispiele

In diesem ersten Beispiel sind zwei Ereignisse definiert, die eintreten, wenn ein Signal an Eingang X0 bzw. X1 angelegt wird. Die Mails, die erzeugt werden, unterscheiden sich durch ihre Betreffzeilen, aber Empfänger und Meldungstext sind in beiden Fällen gleich. Eine Besonderheit hat es noch mit dem Meldungstext „.“ auf sich. Text, der hinter einem solchen einsamen Punkt folgt, wird beim Versand der Mail abgeschnitten, also auch der Statusbericht, der normalerweise an jede Mail angehängt wird (s. Beschreibung zum Konfigurationszweig *report*). Und diesen Statusbericht zu unterdrücken, kann wünschenswert sein, um den Empfänger nicht mit unnötigen Details zu verwirren.

```
<io-mail.1>
  <alarm>
    <event>
      <trigger op="EQ" var="X0" val="1"/>
      <subject>Bitte Kaugummiautomaten nachfüllen</subject>
    </event>
    <event>
      <trigger op="EQ" var="X1" val="1"/>
      <subject>Bitte Zigarettensautomaten nachfüllen</subject>
    </event>
    <!-- die folgenden Angaben gelten für beide event's -->
    <mailto>f.mustermann@firma.de</mailto>
    <message>Wegen Schlüssel beim Hausmeister melden.</message>
    <message>.</message>
  </alarm>
</io-mail.1>
```

Wie bereits erwähnt, lassen sich auch mehrere Triggerbedingungen miteinander verknüpfen. Das ist vor allem sinnvoll im Zusammenhang mit den zeitabhängigen Variablen (WE, NT), deren Änderungen für sich alleine sicher nicht besonders interessant sind:

```
<io-mail.1>
  <alarm>
    <event>
```

```
<!-- Die beiden folgenden Bedingungen müssen gleichzeitig -->
<!-- erfüllt sein (implizite UND-Verknüpfung): -->
<trigger op="EQ" var="X0" val="1"/>
<trigger op="EQ" var="WE" val="1"/>
<subject>Störung in der Heizungsanlage</subject>
<message>Und das ausgerechnet am Wochenende...</message>
</event>
<mailto>f.mustermann@firma.de</mailto>
</alarm>
</io-mail.1>
```

Man kann den IO-Mailer auch dazu bringen, eine Einschaltmeldung zu verschicken, z. B. um zu überwachen, ob und wann vorübergehende Ausfälle der Versorgungsspannung auftreten. Zu dem Zweck formuliert man ein Ereignis ohne jede Triggerbedingung. Ein solches Ereignis tritt dann einmal ein (weil es keine Triggerbedingung gibt, die nicht erfüllt wäre) und danach nie wieder (weil es eben *andauert*, was erneutes Eintreten ausschließt).

```
<io-mail.1>
<alarm>
  <event>
    <mailto>f.mustermann@firma.de</mailto>
    <subject>E/A-Server startet</subject>
  </event>
  ...
</alarm>
</io-mail.1>
```

Eine leichte Abwandlung dieser Technik läßt sich auch verwenden, um in regelmäßigen Abständen Statusberichte zu verschicken. Der *repeat*-Parameter bewirkt in diesem Beispiel, dass das Ereignis, solange seine Triggerbedingungen erfüllt sind (und das sind sie nun mal immer), in Abständen von einer Stunde immer wieder erneut eintritt.

```
<io-mail.1>
<alarm>
  <event>
    <mailto>f.mustermann@firma.de</mailto>
```

```
<subject>Keine besonderen Vorkommnisse</subject>
<message>Nur schon wieder eine Stunde um.</message>
<repeat>01:00:00</repeat>
</event>
</alarm>
</io-mail.1>
```

## 4.10.2 Beispiele zur Benutzung des seriellen Ports

Damit die folgenden Beispiele funktionieren, muß der serielle Port sich im Peripheriemodus (statt Terminalmodus) befinden, für die beiden letzten außerdem im Datenmodus (statt Kommandomodus), d.h. `<packet mode="D">`, nicht „C“.

Das erste Beispiel zeigt das Versenden von seriellen Daten. X1 und X0 sollen für dieses Beispiel jeweils mit einer Lichtschranke oder einem Drehkreuz an der Eingangs- und Ausgangstür eines Raumes verbunden sein, so dass in der Variablen N1 die Personen gezählt werden, die den Raum betreten, und in N0 die, die ihn wieder verlassen. DN schließlich (die Differenz aus N1 und N0) enthält dann die Anzahl Personen, die sich im Raum aufhalten.

Mit der folgenden Konfiguration wird eine Meldung ausgegeben, sobald sich genau zehn Personen im Raum befinden, und danach jedesmal, wenn noch eine weitere hinzukommt. Das dritte Ereignis schließlich bewirkt, dass die Zähler auf Null zurückgesetzt werden, wenn die „Anzahl Personen“ negativ wird, was offensichtlich kein sinnvolles Ergebnis ist.

```
<io-mail.1>
  <alarm>
    <event>
      <trigger op="EQ" var="DN" val="10"/>
      <action var="TX" val="Jetzt sind's genau zehn.\r\n"/>
    </event>
    <event>
      <trigger op="DELTA" var="N1" val="1"/>
      <trigger op="GT" var="DN" val="10"/>
      <action var="TX" val="Da kommt noch einer! „"/>
```

```
    <action var="TX" val="So langsam wird's voll hier!\r\n"/>
  </event>
<event>
  <trigger op="LT" var="DN" val="0"/>
  <action var="N0" val="0"/>
  <action var="N1" val="0"/>
  <action var="TX" val="Hoppla, erzählt...\r\n"/>
</event>
</alarm>
</io-mail.1>
```

Besondere Beachtung verdient dabei das zweite Ereignis, das zwei Aktionen mit Zuweisungen an TX enthält. So etwas ist erlaubt und bewirkt, dass die Texte nacheinander ausgegeben werden, in der Reihenfolge, wie sie in den Konfigurationsdaten stehen. Längere Texte lassen sich auf diese Weise in bis zu 15 Abschnitte (z.B. Zeilen) unterteilen. Außerdem noch zu sehen: Anders als bei Mail-Texten hat man bei seriellen Ausgaben volle Kontrolle über jedes einzelne ausgegebene Zeichen. Insbesondere der Zeilenumbruch muß explizit angegeben werden, in diesem Fall mit den Steuerzeichen „\r\n“ für CR, LF.

Das nächste Beispiel besteht eigentlich aus zwei Teilen. Das erste Ereignis enthält eine serielle Triggerbedingung, die den Merker M0 setzt, sobald ein Paket mit dem Text „Alarm“ empfangen wurde, und über einen Taster an Eingang X0 läßt sich M0 wieder zurücksetzen. Stellen Sie sich jetzt noch vor, dass Ausgang Y0 ein Lämpchen ansteuert. Und angenommen, die beiden ersten Ereignisse würden direkt Y0 ansteuern statt M0, wäre man an dieser Stelle bereits fertig. Stattdessen bewirken hier die drei übrigen Ereignisse, dass dieses Lämpchen blinkt (0.3 Sek. ein, 0.2 Sek. aus), solange M0 gesetzt ist.

```
<io-mail.1>
  <alarm>
    <!-- M0 setzen/zurücksetzen -->
  <event>
    <dtrigger substring="alarm" case="0"/>
    <action var="M0" val="1"/>
```

```
</event>
<event>
  <trigger op="EQ" var="X0" val="1"/>
  <action var="M0" val="0"/>
</event>
<!-- Y0 blinkt, solange M0 gesetzt ist -->
<event>
  <trigger op="EQ" var="M0" val="1"/>
  <trigger op="EQ" var="Y0" val="0" delay="0.2"/>
  <action var="Y0" val="1"/>
</event>
<event>
  <trigger op="EQ" var="M0" val="1"/>
  <trigger op="EQ" var="Y0" val="1" delay="0.3"/>
  <action var="Y0" val="0"/>
</event>
<!-- Y0 aus bei M0=0 -->
<event>
  <trigger op="EQ" var="M0" val="0"/>
  <action var="Y0" val="0"/>
</event>
</alarm>
</io-mail.1>
```

Das letzte Beispiel zeigt die Verknüpfung mehrerer serieller Trigger untereinander und außerdem noch mit einer regulären Triggerbedingung. Einige Besonderheiten sind auch im *message*-Text enthalten: Der Platzhalter %s wird beim Versenden der Mail durch den Inhalt des Datenpakets ersetzt, das das Ereignis ausgelöst hat, und das Steuerzeichen \n erzwingt einen Zeilenumbruch. Dass durch einen einzigen Punkt in der letzten Zeile das Anhängen des Gerätestatus verhindert werden kann, wurde bereits oben erwähnt. Neu ist, dass der Punkt keine eigene *message* sein muß, wenn ihm wie hier ein \n vorausgeht.

```
<io-mail.1>
  <alarm>
    <event>
      <!-- drei implizit UND-verknüpfte Triggerbedingungen: -->
      <dtrigger substring="alarm"/>
```



```
<dtrigger substring=" 51 #"/>
<trigger op="EQ" var="NT" val="0"/>
<mailto>f.mustermann@firma.de</mailto>
<subject>Alarmmeldung von Station 51</subject>
<message>Der Meldungstext lautet:\n%s\n.</message>
</event>
</alarm>
</io-mail.1>
```

Zur Funktionsweise der `<dtrigger>` ist noch zu sagen, dass sie entweder sofort ein Ereignis auslösen oder nie. Das bedeutet, dass die beiden gesuchten Texte in ein- und demselben Paket vorkommen müssen, um etwas zu bewirken. Über mehrere Pakete verteilte Treffer zählen nicht, egal wie kurz der Abstand zwischen den Paketen wäre. Und Pakete, die zwar beide Texte enthalten, aber während der Nacht eintreffen (wenn also die Variable NT ungleich 0 ist), bewirken auch nichts. Insbesondere nicht dass, wie man vielleicht annehmen könnte, eine Mail gesendet wird, sobald die Nacht endet.

## 5 Einschränkungen

### 5.1 Was nicht geht

- Es ist nicht möglich, als Triggerbedingung zwei Variablen miteinander zu vergleichen., immer nur eine Variable mit einem konstanten Wert. Lediglich ein Vergleich zwischen den Zählern N0 und N1 ist (indirekt) möglich, nämlich über den Inhalt von DN.
- Solange der serielle Port im Kommandomodus ist, <packet mode="C"/>, bleiben über <dtrigger/> angegebene serielle Triggerbedingungen wirkungslos.
- Der Platzhalter %s bewirkt nur im *message*-Text einer Mail etwas. Es ist nicht möglich, das empfangene serielle Paket in die Betreffzeile der Mail oder in serielle Ausgaben einzufügen.

### 5.2 Grenzwerte

Einige dieser Zahlen wurden bereits anderswo in der Anleitung erwähnt und sind hier nur noch einmal zur Übersicht aufgeführt.

- In der vorliegenden Firmware-Version können maximal 15 Ereignisse, 25 Triggerbedingungen und 40 Aktionen definiert werden.
- Die größtmöglichen Werte für Ansprechverzögerung und Erholzeit einer Triggerbedingung sind 00:10:55, also knapp elf Minuten.
- Meldungstexte können höchstens 254 Zeichen lang sein (das entspricht bei Mail ca. 3 - 4 Zeilen). Der Inhalt einer Mail kann zwar aus mehreren *message*-Blöcken bestehen, diese Texte werden aber nicht nahtlos aneinandergehängt, sondern jeder wird als eigener Absatz dargestellt. Außerdem steht für alle Konfigurationsdaten zusammen nur 8 KByte Speicher zur Verfügung.
- Die seriellen Ausgaben zu einem Ereignis können aus max. 15 Zuweisungen an TX bestehen.

- Der serielle Empfangspuffer ist 8 KByte groß. Ein einzelnes serielles Paket, das diese Größe überschreitet, wird darum zwangsläufig abgeschnitten.
- Das XML-Dokument mit den Konfigurationsdaten darf höchstens 20 KByte lang sein, egal, ob es per Mail oder lokal über den seriellen Port eingereicht wird. Kommentare zählen dabei nicht, an denen brauchen Sie also nicht unbedingt zu sparen. Allerdings besteht für Mails darüber hinaus eine Beschränkung auf 64 KByte Gesamtgröße, noch größere werden ungelesen gelöscht. Diese Grenze ist relativ willkürlich gesetzt, aber grundsätzlich notwendig, da der IO-Mailer sonst durch (egal ob böswillig oder irrtümlich gesendete) riesige Dateianhänge über lange Zeiträume blockiert werden könnte.

### 5.3 Diverse mögliche Probleme

- Da es oftmals möglich ist, einen Mail-Empfänger unter mehreren Varianten ein- und derselben Adresse zu erreichen (z.B. „f.mustermann@mail.firma.de“ als Äquivalent zu „f.mustermann@firma.de“), kann es sein, dass nicht alle Empfänger das „unsubscribe“-Kommando verwenden können. Es wird für solche Empfänger fehlschlagen, die in der Konfiguration unter einer anderen Adresse eingetragen sind als der, die ihre eigene Mail-Software als Absenderadresse vermerkt.
- Anders als in der XML-Spezifikation verlangt, versteht der IO-Mailer nicht die 16bit-Zeichensatz-Codierungen UTF-8 und UTF-16, sondern nur den 8bit-Zeichensatz ISO-8859-1 sowie als Teilmenge davon 7bit-US-ASCII. Da ISO-8859-1 der in Westeuropa übliche Zeichensatz unter Windows und Unix ist, wird diese Einschränkung normalerweise keine Probleme bereiten, solange Sie einen einfachen Texteditor zum Bearbeiten der Konfigurationsdaten verwenden. Wenn Sie aber einen speziellen XML-Editor benutzen, sollten sie darauf achten, dass Sie die Daten keinesfalls als UTF-16 speichern, stattdessen wenn möglich als ISO-8859-1, oder notfalls als UTF-8.
- Der IO-Mailer versteht nicht das MIME-Codierverfahren base64, sondern nur quoted-printable und darüber hin-

aus natürlich einfachen Klartext. Vernünftige Mail-Software sollte keinen Anlaß sehen, die Konfigurationsdaten jemals als base64 zu codieren. Falls das aber dennoch geschieht, werden die Konfigurationsdaten mit der Fehlermeldung „start tag <io-mail.1> not found“ abgelehnt werden. Sie können in so einem Fall vielleicht noch versuchen, das Verhalten Ihrer Mail-Software zu beeinflussen, indem Sie die Konfigurationsdaten nicht als Dateianhang schicken, sondern direkt in den Textkörper der Mail einfügen.

### 5.4 IO-Mailer und Exchange Server 5.5

Microsoft Exchange Server und Microsoft Exchange-Clients verwenden zur Kommunikation nicht SMTP und POP3, sondern das MAPI-Protokoll. Um den Missbrauch des Servers durch Dritte zu verhindern sind Exchange Server oft so konfiguriert, dass ein Zugriff via SMTP den E-Mail-Versand nur an Empfänger im Eigenen LAN zulässt.

Damit der IO-Mailer auch im Zusammenspiel mit Exchange Servern uneingeschränkt E-Mails versenden kann muss der Exchange konfiguriert werden, so dass unerwünschtes Mail-Relay (d. h. Missbrauch durch Spammer) verhindert wird, Mail-Relay für ausgewählte Clients aber erlaubt bleibt:

Die betreffenden Einstellungen befinden sich in den Eigenschaften des Internet-Mail-Dienstes, Registerkarte „Routing“.

Zu Zeiten von Exchange 5.0 war empfohlen, hier „Eingehende Nachrichten nicht umleiten“ auszuwählen. Die Maßnahme betrifft aber alle per SMTP eingelieferten Mails, also auch solche, die z. B. der I/O-Mailer zu senden versucht.

Seit Exchange 5.5 Service Pack 2 gibt es hier eine zusätzliche Schaltfläche „Routingbeschränkungen“, mit der es möglich ist, feiner unterteilte Richtlinien festzulegen.

Dazu:

- Doch die Option „Eingehende SMTP-Nachrichten umleiten“ wählen.
- Zur Routing-Liste den oder die Namen der eigenen Mail-Domäne hinzufügen, mit der Eigenschaft „als inbound akzeptieren“.
- „Routingbeschränkungen“ klicken und als anzuwendendes Filterkriterium „Hosts und Clients mit folgenden IP-Adressen“ auswählen. In die zugehörige Liste die IP-Adressen oder IP-Netze eintragen, denen SMTP-Relay erlaubt werden soll. Im einfachsten Fall wäre das das eigene LAN, also z. B. etwas wie 172.16.232.0, Mask 255.255.255.0

## 6 Internet-Adressen

### 6.1 Zum Thema XML

Allgemeines über XML (Spezifikation, Fragen und Antworten):

<http://www.w3.org/TR/REC-xml>

<http://msdn.microsoft.com/xml/tutorial/default.asp>

<http://www.ucc.ie/xml/>

<http://msdn.microsoft.com/workshop/xml/general/xmlfaq.asp>

Gratis-Download: Der Microsoft XML-Validator (benötigt Internet Explorer 5). Was man hier erhält, ist im wesentlichen eine HTML-Seite mit ein wenig eingebettetem JavaScript, das nichts weiter tut, als den Namen einer XML-Datei einzulesen und diese zur Überprüfung an die eingebauten Routinen des Explorers weiterzureichen. Sehr hilfreich, wenn man XML-Dateien bereits vor Versenden an den IO-Mailer auf korrektes Format prüfen will.

[http://msdn.microsoft.com/downloads/samples/internet/xml/xml\\_validator/default.asp](http://msdn.microsoft.com/downloads/samples/internet/xml/xml_validator/default.asp)

### 6.2 Zum Thema Timeserver

Verzeichnis weltweit erreichbarer Timeserver im Internet, eigentlich: NTP-Server. NTP ist ein aufwendiges Protokoll zur möglichst präzisen Uhrensynchronisation in weitläufigen Netzwerken (s. RFC-1305), was nicht dasselbe ist wie der einfache Zeitansagedienst nach RFC-868, den der IO-Mailer benötigt. In der Praxis bieten die meisten NTP-Server aber auch den einfachen Zeitansagedienst zusätzlich an.

Beachten Sie bitte, dass Sie nur solche Server aus dieser Liste verwenden sollten, die ausdrücklich als „Open access“ gekennzeichnet sind.

<http://www.eecis.udel.edu/~mills/ntp/servers.htm>

Gratis-Download: Timeserver für Windows-Rechner. Für die meisten Zwecke dürfte eine solche lokale Lösung sinnvoller sein, als auf einen fremden Timeserver im Internet zuzugreifen.

<http://www.arachnoid.com/abouttime/index.html>

Wie bereits weiter oben erwähnt, können Sie sich das alles aber sparen, solange Sie irgendeinen eigenen Unix- oder Linux-Server besitzen. Dort ist ein Timeserver nach RFC-868 nämlich bereits eingebaut, und Sie müssen lediglich sicherstellen, dass er auch aktiviert ist (in der Datei „inetd.conf“ der Dienst namens „time“ auf TCP-Port 37).

## 7 Firmware-Update

Der IO-Mailer ist intern in die zwei Funktionsbereiche TCP/IP-Ethernet und SMTP/POP3-Client unterteilt.

Jeder der beiden Funktionsbereiche verfügt über eine eigenständige Firmware, die durch ein unabhängiges Update erneuert werden kann.

### 7.1 Update des TCP/IP-Ethernet-Bereichs

#### 7.1.1 unter Windows 9x/NT/2000

Voraussetzung ist ein PC unter Windows mit einem Netzwerkananschluß und aktiviertem TCP/IP-Stack. Für den Update-Prozeß benötigen Sie drei Files, die auf der Website <http://www.wut.de> zum Download bereitstehen.

- Das ausführbare Update-Tool für die Übertragung der Firmware in den IO-Mailer.
- Die Datei mit der neuen Firmware die in den IO-Mailer übertragen werden soll.
- Die Konfigurations-Datei (\*.cfg), die den TCP/IP-Stack auf die vom IO-Mailer benötigten Einstellungen setzt.

Der Updateprozeß ist im folgenden in Einzelschritten erläutert. Bitte beachten Sie auch evt. auf unseren Webseiten veröffentlichte ergänzende Hinweise.

1. Starten Sie eine Telnet-Sitzung auf den Konfigurationsport des Com-Servers.

```
telnet [IP-Adresse] 1111
```

Wählen Sie im Menü *SETUP: System* → *Flash Update* → *Net Update*, und bestätigen Sie mit „y“. Die Netzwerk-Verbindung wird daraufhin vom IO-Mailer geschlossen und die entsprechende Telnet-Meldung muß mit „OK“ bestätigt werden. Die grüne Status-LED zeigt an, dass er sich jetzt im Update-Mode befindet.



2. Starten Sie jetzt das Update-Tool. Über den Menüpfad *CS programming* → *Flash* gelangen Sie in die Eingabemaske für den Upload einer neuen Firmware.
3. Geben Sie in die entsprechenden Felder die IP-Adresse des IO-Mailers sowie die Namen der Firmware-Datei und der Konfigurations-Datei ein. Im Options-Feld „*Output*“ aktivieren Sie bitte ausschließlich den Punkt „*Firmware*“.
4. Klicken Sie auf den Button *Start*. Das Update dauert einige Sekunden. Es ist erst beendet, wenn ein Message-Fenster das Ende des Update-Prozesses meldet.
5. Kontrollieren Sie im Konfigurationsmenü des IO-Mailers, ob die neue Betriebssoftware übernommen wurde. Im Menü *INFO System* → *SOFTW Date/Rev* muß die neue Versionsnummer der Firmware stehen.

Wird hier immer noch die vorherige Version angezeigt, ist das File mit der neuen Betriebssoftware beschädigt. Setzen Sie sich bitte mit Ihrem Händler in Verbindung.

### 7.1.2 unter UNIX

Voraussetzung ist ein Rechner mit einem Netzwerkanschluß und einem TCP/IP-Stack, der die Netzwerkprotokolle Telnet und TFTP zur Verfügung stellt.

Für den Update mit Hilfe des TFTP-Protokolls, können Sie auf Anfrage bei W&T (info@wut.de) die Firmware in 4 Dateien gesplittet bekommen, die nacheinander an den IO-Mailer übertragen werden.

1. Starten Sie das Fernkonfigurationstool des IO-Mailers über Telnet.  

```
telnet [IP-Adresse] 1111
```

Wählen Sie im Menü *SETUP: System* → *Flash Update* → *Net Update*, und bestätigen Sie mit „y“. Die Telnet-Verbindung wird vom IO-Mailer geschlossen. Die grüne Status-LED zeigt an, dass er sich jetzt im Update-Mode befindet.
2. Übertragen Sie nun mit dem Befehl TFTP im Binär-Modus das erste File an den IO-Mailer. Während die Daten über das Netzwerk übertragen werden, blitzt die

Status-LED auf. Danach geht der IO-Mailer in den Programmiermodus und die Fehler-LED leuchtet auf. Dieser Prozeß kann einige Sekunden dauern. Warten Sie, bis die Fehler-LED ausgeht und die Status-LED wieder leuchtet.

Wiederholen Sie diesen Prozeß für alle vier Files.

3. Der IO-Mailer erkennt, wann alle Files übertragen wurden und führt selbständig einen Neustart durch. Sollte nach der Übertragung aller Files wieder die grüne Status-LED leuchten, wiederholen Sie Punkt 2 vollständig. Die doppelte Übertragung eines Files erzeugt keinen Update-Fehler. Der IO-Mailer wartet so lange, bis alle notwendigen Files übertragen wurden.

Beispiel: SCO UNIX

Geben Sie die folgenden Befehle nach dem jeweiligen Prompt ein:

```
# tftp
tftp> connect [ip_number|host_name]
tftp> binary
tftp> put C4r1_1.4_1 [remote filename]
      (remote filename = irgendein Buchstabe)
```

Warten Sie jetzt, bis die grüne Status-LED wieder leuchtet. Übertragen Sie dann mit dem Befehl *put* die restlichen drei Files und beenden die TFTP-Verbindung.

```
tftp> quit
#
```

4. Kontrollieren Sie im Konfigurationsmenü des IO-Mailers, ob er die neue Betriebssoftware übernommen hat. Im Menü *INFO Com-Server* → *SOFTW Date/Rev* muß nun die Versionsnummer der neuen Firmware erscheinen.

Wird nach wie vor die bisherige Version angezeigt, sind eine oder mehrere der Dateien mit der neuen Betriebssoftware

beschädigt. Setzen Sie sich bitte mit unserer Hotline in Verbindung.

## 7.2 Update des SMTP/POP3-Cient-Bereichs

Für den Update-Prozeß benötigen Sie ein Update-File (\*.mhx), das auf der Website <http://www.wut.de> zum Download bereitsteht.

### 7.2.1 Update über das Netzwerk

1. Öffnen Sie die das mxh-File in einem Editor.
2. Kopieren Sie den kompleten File-Inhalt in die Zwischenablage (*Bearbeiten* → *Alles markieren* → *Bearbeiten* → *kopieren*).
3. Jetzt muß der IO-Mailer in den Updatemodus gebracht werden. Hierfür gibt es zwei Möglichkeiten:
  - Senden einer E-Mail mit dem Betreff Update an den IO-Mailer
  - Verbinden Sie den seriellen Port des IO-Mailers mit einem COM-Port Ihres PC.  
Öffnen Sie ein Terminalprogramm mit 9600Baud, 8 Bit, no Parity, no Handshake.

Bringen Sie den IO-Mailer in den Terminal-Mode indem Sie bei Zuführen der Versorgungsspannung mehrere kleine „x“ tippen, bis im Terminalprogramm IP+Enter erscheint. Drücken Sie die Enter-Taste und dann die #-Taste.

Es erscheint:

```
MB90F562 bootloader v1.6 W&T 06/2001
Invoked by software, ESC to cancel
Waiting (Port 1)...
```

4. Starten Sie eine Telnet-Sitzung auf den IO-Mailer

```
telnet [IP-Adresse]
```

5. Fügen Sie nun den Inhalt der Zwischenablage unter Telnet ein (*Bearbeiten* → *Einfügen*)

Es erscheint nacheinander

```
Programming...  
48074  
Done!
```

Nach Einspielen der neuen Firmware beendet der IO-Mailer automatisch die Telnet-Sitzung.

## 7.2.2 Update über die serielle Schnittstelle

1. Öffnen Sie die das mxh-File in einem Editor.
2. Kopieren Sie den kompleten File-Inhalt in die Zwischenablage (*Bearbeiten* → *Alles markieren* → *Bearbeiten* → *kopieren*)
3. Jetzt muß der IO-Mailer in den Updatemodus gebracht werden. Verbinden Sie den seriellen Port des IO-Mailers mit einem COM-Port Ihres PC.  
Öffnen Sie ein Terminalprogramm mit 9600Baud, 8 Bit, no Parity, Handshake XON/XOFF  
Bringen Sie den IO-Mailer in den Terminal-Mode indem Sie bei Zuführen der Versorgungsspannung mehrere kleine „x“ tippen, bis im Terminalprogramm *IP+Enter* erscheint.

Drücken Sie die Enter-Taste und dann die #-Taste.

Es erscheint:

```
MB90F562 bootloader v1.6 W&T 06/2001  
Invoked by software, ESC to cancel  
Waiting (Port 0)...
```

4. Fügen Sie nun den Inhalt der Zwischenablage im Terminal-Programm ein (*Bearbeiten* → *Einfügen*)

Es erscheint nacheinander

```
Programming...  
48074  
Done!
```

## **W&T**

Damit ist das Firmware Update abgeschlossen.

**Technische Daten Typen 57111, 57211**

	57111	57211
Spannungsversorgung:	5V über Steckernetzteil 230V/50Hz	
Stromaufnahme:	typisch 270mA max. 330mA	typisch 520mA max. 630mA
Konfigurationsschnittstelle:	RS232 DCE-Belegung auf DB9F 9600Bd, 8 Bit, no Parity (nur RxD und TxD unterstützt)	
Digitale Eingänge:	2 Eingänge, spannungsfest bis +/-30V CMOS-kompatible Schaltschwelle mit 32-Bit-Zählfunktion (max. +/-2147483647)	
Digitale Ausgänge:	2 Ausgänge „Open Collector“ max. 30V/100mA	
Netzwerk:	Ethernet 10BaseT	Ethernet 100BaseTx/10BaseT
Mailfunktionen:	Konfiguration, Alarmierung, Berichtsfunktionen Setzen der Ausgänge	
Mailprotokolle:	POP3, SMTP	
Gehäuse:	Kunststoffkleingehäuse, 105x75x22mm geeignet für Hutschienenmontage	
Gewicht:	ca. 200g	
Umgebungstemperatur Betrieb:	0..+50°C bei angereichten Gehäusen 0..+60°C bei Einzelbetrieb	
Umgebungstemperatur Lagerung:	-40..+70°C	

**Technische Daten Typen 57511, 57611**

	57511	57611
Spannungsversorgung:	12–24V AC oder DC über Schraubklemme	
Stromaufnahme:	typisch 155mA @24V max. 205mA @24V	max. 175mA @24 typisch 225mA @24V
Konfigurationsschnittstelle:	RS232 DCE-Belegung auf DB9F 9600Bd, 8 Bit, no Parity (nur RxD und Txd unterstützt)	
Digitale Eingänge:	2 Eingänge, spannungsfest bis +/-30V CMOS-kompatible Schaltschwelle mit 32-Bit-Zählfunktion (max. +/-2147483647)	
Digitale Ausgänge:	2 Ausgänge „Open Collector“ max. 30V/100mA	
Netzwerk:	Ethernet 10BaseT	Ethernet 100BaseTx/10BaseT
Mailfunktionen:	Konfiguration, Alarmierung, Berichtsfunktionen Setzen der Ausgänge	
Mailprotokolle:	POP3, SMTP	
Gehäuse:	Kunststoffkleingehäuse, 105x75x22mm geeignet für Hutschienenmontage	
Gewicht:	ca. 200g	
Umgebungstemperatur Betrieb:	0..+50°C bei angereichten Gehäusen 0..+60°C bei Einzelbetrieb	
Umgebungstemperatur Lagerung:	-40..+70°C	

**EG-Konformitätserklärung**

nach Artikel 10.1 der Richtlinie 89/336/EWG

Die Wiesemann & Theis GmbH, Wuppertal erklärt, dass die Produkte

W&T IO-Mailer Typ 57111, Typ 57211

W&T IO-Mailer Typ 57511, Typ 57611

auf die sich diese Erklärung bezieht, mit den folgenden Normen bzw. normativen Dokumenten übereinstimmen:

1. EN 55022-B (1998)  
Produktspezifische Emissionsnorm  
für Kommunikationstechnik
2. EN 50082-2 (1996)  
Störfestigkeit
  - 2.1. EN 61000-4-2 ESD
  - 2.2. EN 61000-4-3 Einstrahlung E-Feld
  - 2.3. EN 61000-4-4 Burst
  - 2.4 EN 61000-4-5 Surge
  - 2.5. EN 61000-4-6 Einströmung
  - 2.6. EN 61000-4-8 Einstrahlung Magnetfeld  
(nicht anwendbar)
  - 2.7. EN 61000-4-11 Spannungsunterbrechung
3. EN 60950 (1997)  
Produktspezifische Niederspannungsrichtlinie  
für Kommunikationstechnik

Wuppertal, den 13.09.01



Klaus D. Meyer  
Vertriebsleiter



Dipl.-Ing. Rüdiger Theis  
Geschäftsführer



Anhang: Optionales Zubehör

Schraubklemmadapter Typ 50883

Zur einfachen Anbindung von digitalen Signalen, ist der Schraubklemmadapter Typ 50883 erhältlich.



Beschaltungsbeispiel:

