essec
telecom systems

System description
Netrix Message Server

Netrix
CROSS MESSAGE SERVER

essec
telecom systems

Table of content

1. Purpose

With this server it is possible to call up messages of any type. These may originate from any type of device.

2. Environment

The software is written for a Windows environment. The program code was written in Visual Basic .NET, and all files are stored in XML format so there is no need for any database software. A certain type of industrial computer is used for the hardware in order to maintain a high level of reliability. We recommend Windows XP as operating system. Multiple Netrix servers can be connected together via a data network.

3. Computer types

3.1 Small Box Type I (No longer available!)

- Has 4 COM ports

3.2 Small Box Type II (in use since September 2005)

- Has 4 COM ports
- 220 V power supply (direct)
- Has one PCI slot, in which we usually install a modem

3.3 Small Box Type III (in use since April 2009)

- Has 4 COM ports
- 220 V power supply or 24 V power supply
- Has one PCI slot, in which we usually install a modem

3.4 Industrial PC

- This computer has 2 COM ports on the motherboard
- There is an extra Moxa card in the system, which provides 8 additional COM ports
- 220 V power supply or 24 V power supply
- May have a built-in sound card
- Internal or external modem

4. How to establish connection

- By making use of the attached keyboard, mouse and screen.
- Via a dial-in RAS connection through which you ‘take over’ the PC using the Dameware software.
- Via a LAN connection through which you ‘take over’ the PC using the Dameware software.
- By establishing a direct network connection via a crossover cable and then starting Dameware.

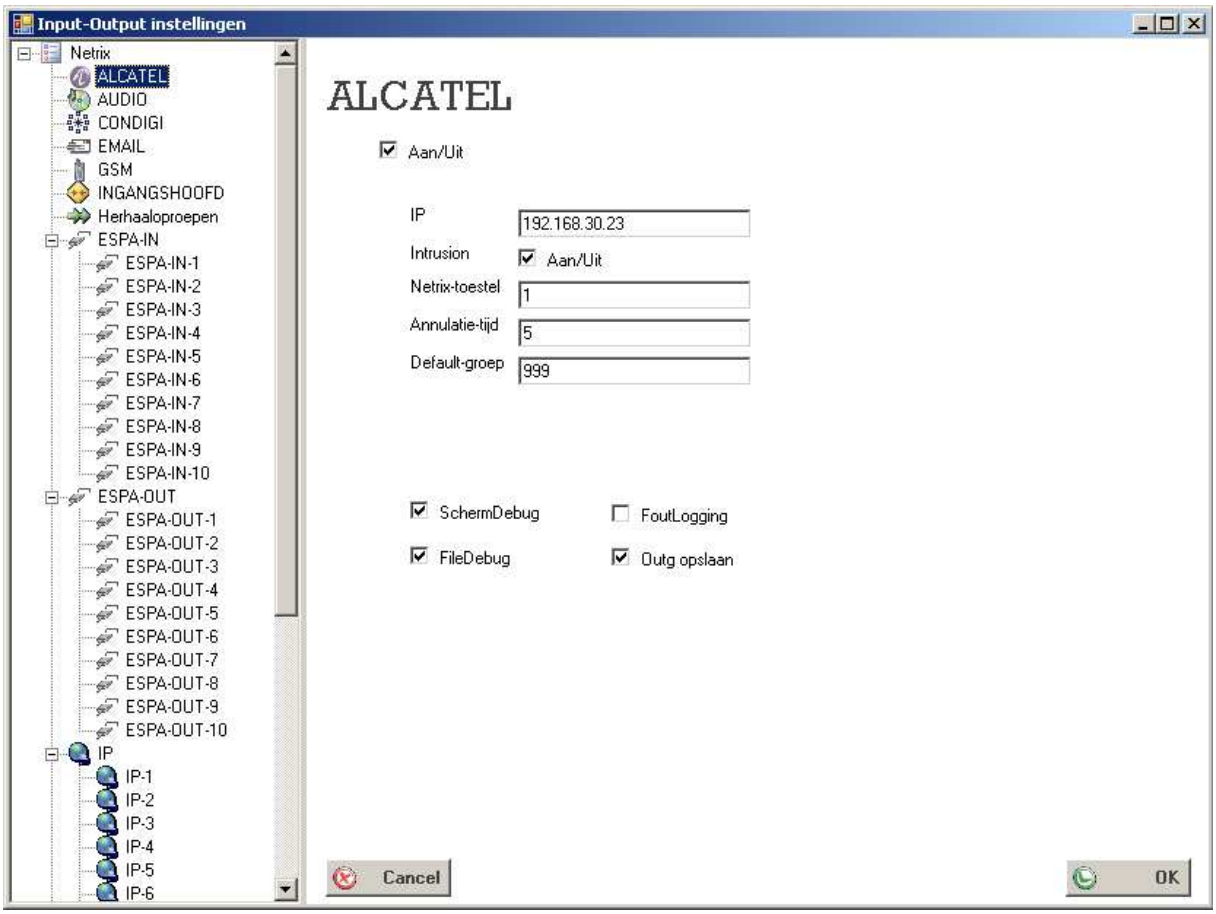
5. Licence

The licence is tied to the computer. If you transfer a database from computer A to computer B, computer B will require its own licence. This cannot be transferred.
The request for a new licence can be submitted to Peter Jans.

The software can also be started without a licence by adding /ESSEC to the command line. This allows you to use the software with all the licences for 24 hours; the software then returns to the installed licences.

6. Inputs and outputs

You can find an overview of the inputs and outputs using the menu option Setup -> Input-Output.
You will find all the inputs and outputs for which you have licences in the tree structure on the left side. Click an item in the tree structure to change the settings.



6.1 Inputs

The currently available inputs are:

- RS232 ports
- Contact inputs, 320 maximum (10x 32); these are provided via the Philips contact heads.
- ESPA inputs
- Incoming SMS message
- IP
- Condigi IP receiver

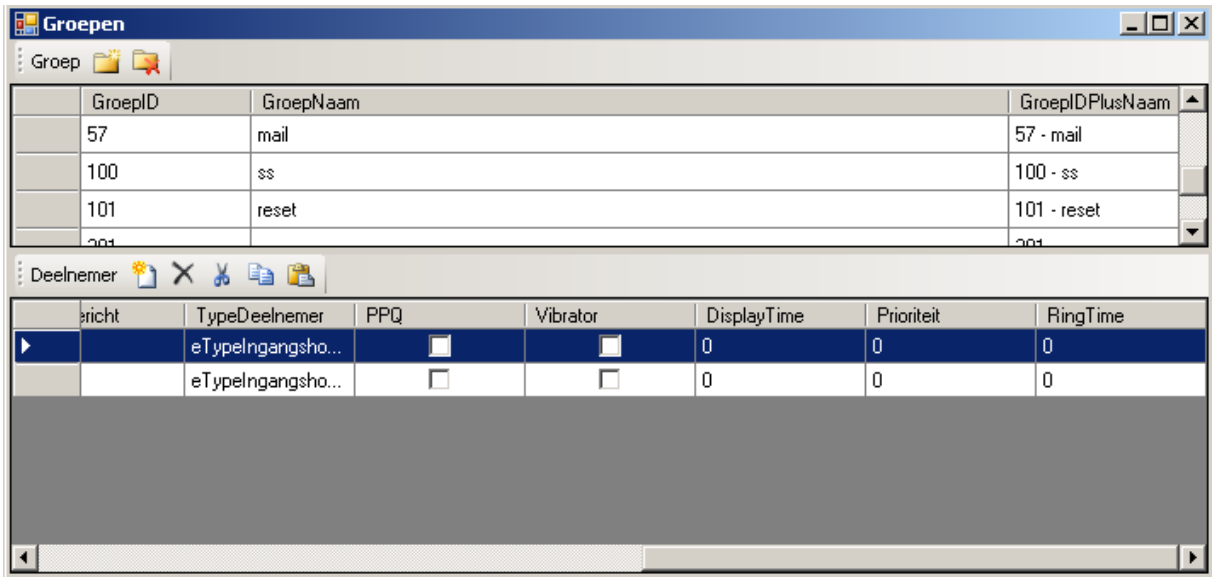
6.2 Outputs

The currently available outputs are:

- Message to mobile phone
- Message to Idect, Ditel telephone system
- Message to Audio, with contact for telephone or PA
- Message to a different group
- Message to a text file
- Message to an RS232 port
- To a group on another Netrix via the network
- Contact head output contacts
- A short e-mail message
- Start repeat call
- Stop repeat call
- Execute command
- Message to Alcatel telephone system
- ESPA-Out message

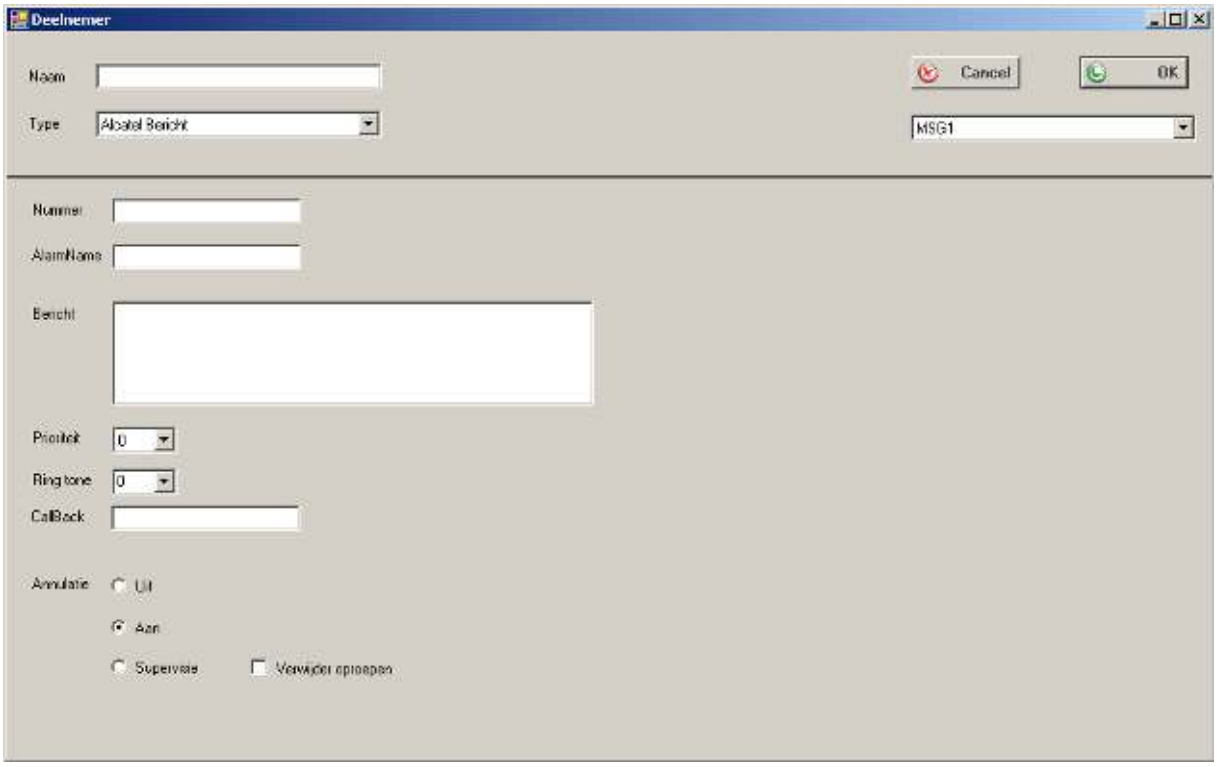
7. Groups

This system allows you to program a maximum of 10,000 groups, and each group can have an unlimited number of members. Every member can be of the type described in the outputs. For each member you can assign a different setting, text and other properties.



8. Members

8.1 Alcatel output



Number: internal number of an Alcatel phone

Alarm name: Title used for message

Priority:

0= Normal call, no acknowledge
1= Normal call with acknowledge
2=Urgent call, no acknowledge
3=Urgent call with acknowledge

Ringtone: each value produces a different rhythm of ringtone

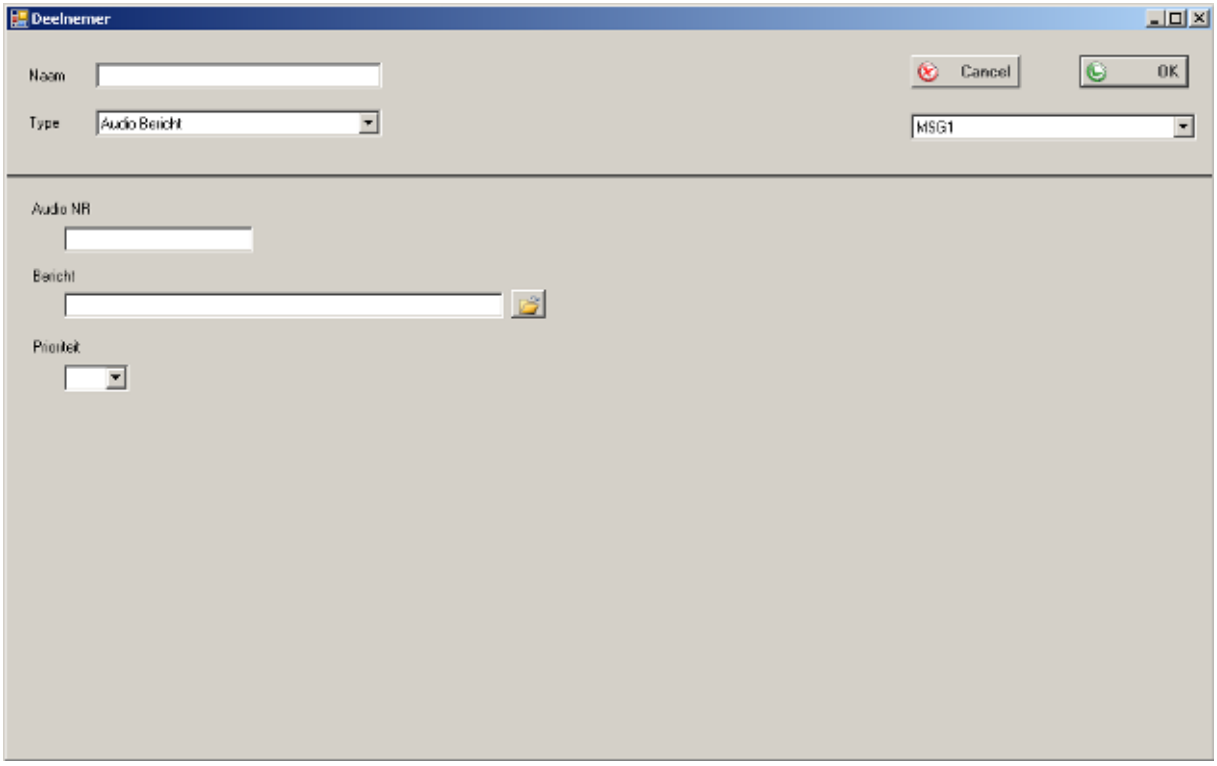
0= ringtone
1= ringtone
2= ringtone
3= ringtone
4= ringtone + vibration
5= ringtone

Callback: Number to use for intrusion.

Cancellation:

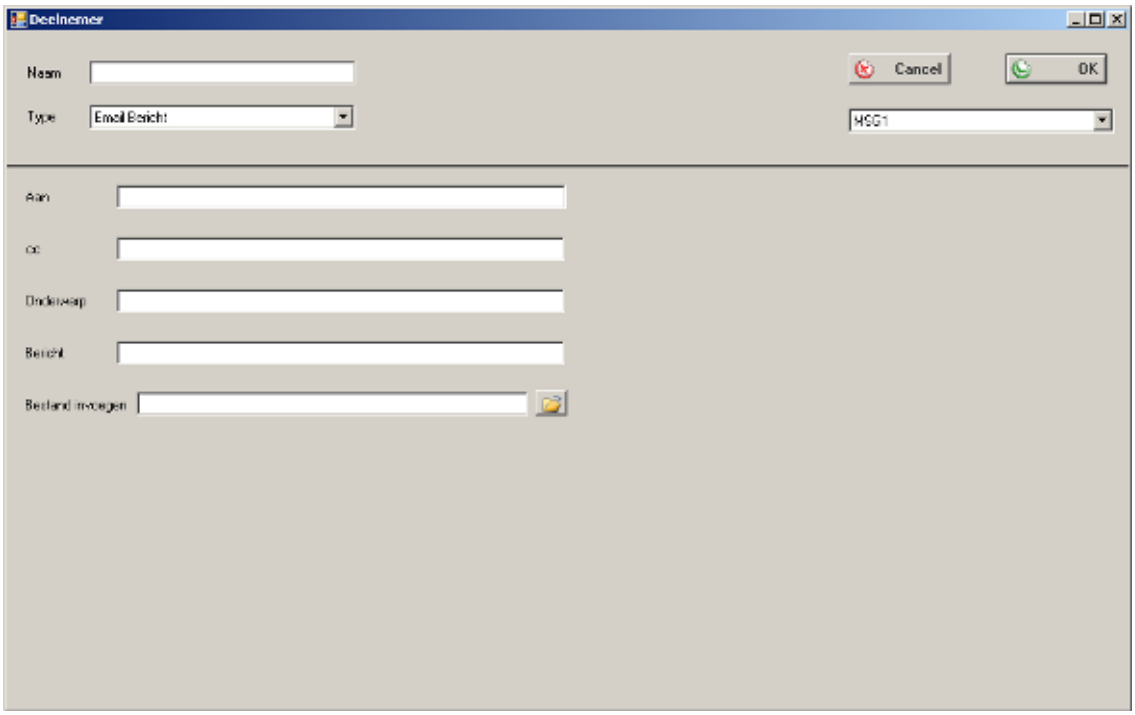
Off = phone continues to ring until message is read.
On = phone rings for configured number of seconds.
Supervision = phone continues to ring until message is read/confirmed.
Cancel calls = If multiple phones receive the same message, as soon as one person reads/confirms the message the calls to other phones are cancelled.

8.2 Audio output

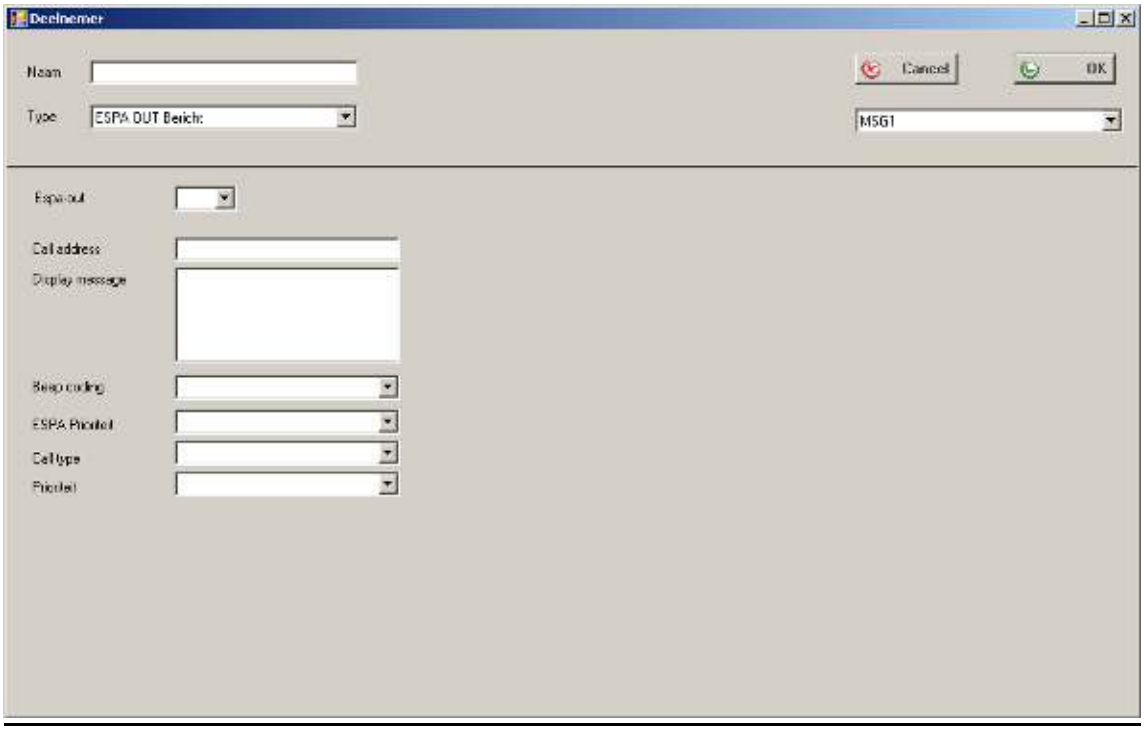


Audio number: telephone number
Message: Location of the file.

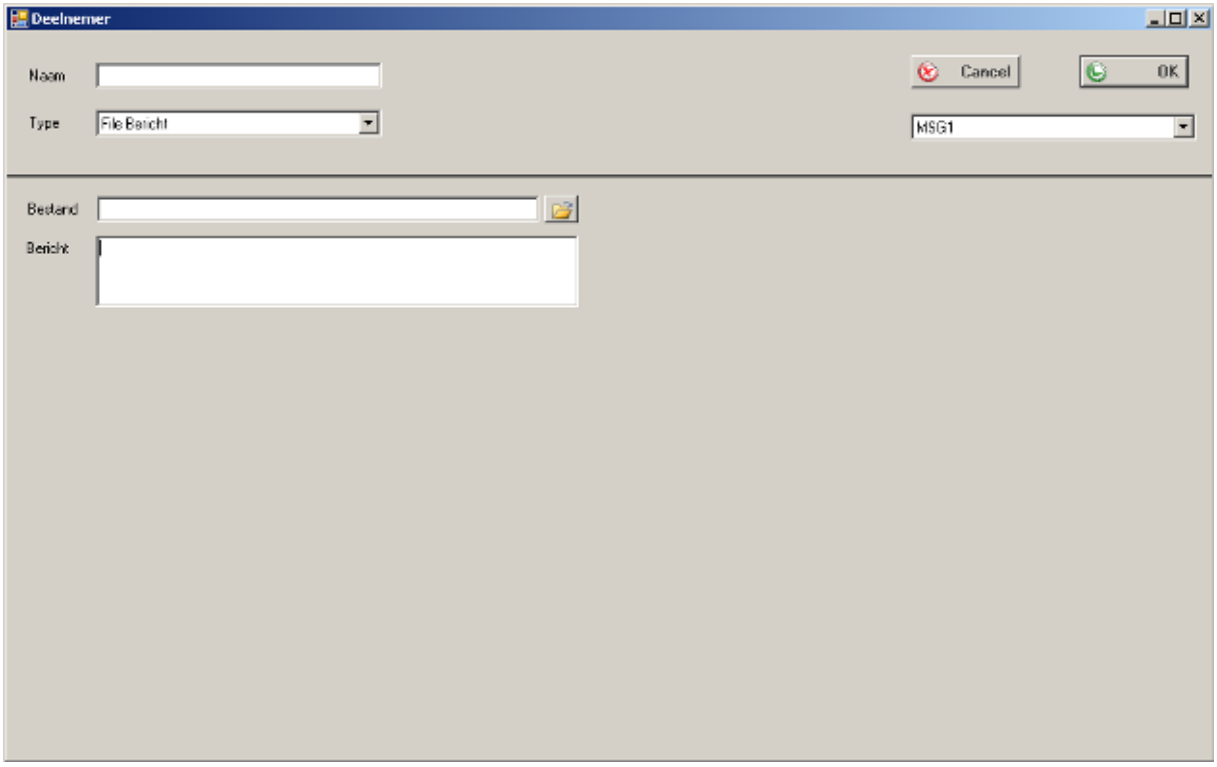
8.3 E-mail message



8.4 ESPA-Out

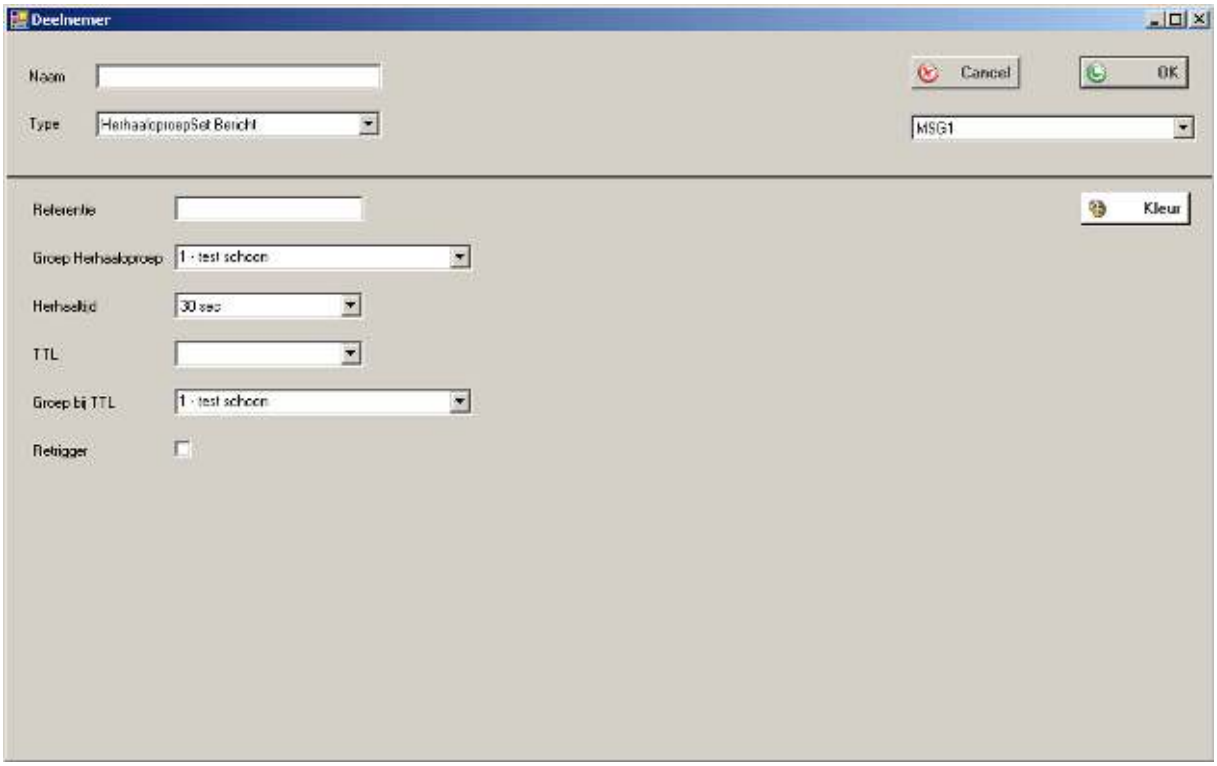


8.5 File



File: Location of the file.

8.6 Repeat call



Reference: unique ID for each call (e.g. pendant number / room number)

Group repeat call: Groups that are called

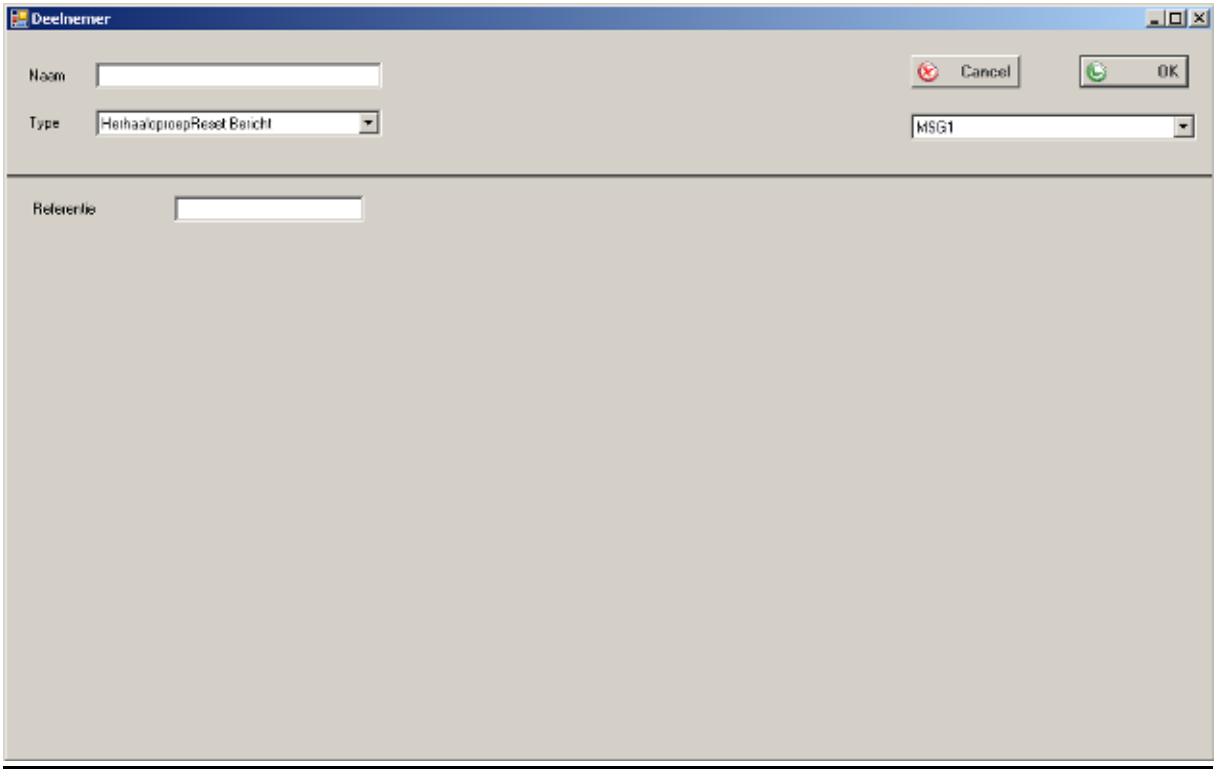
Repeat time: Length of time to wait between the calls.

TTL: number of times a call is repeated. (Time to live)

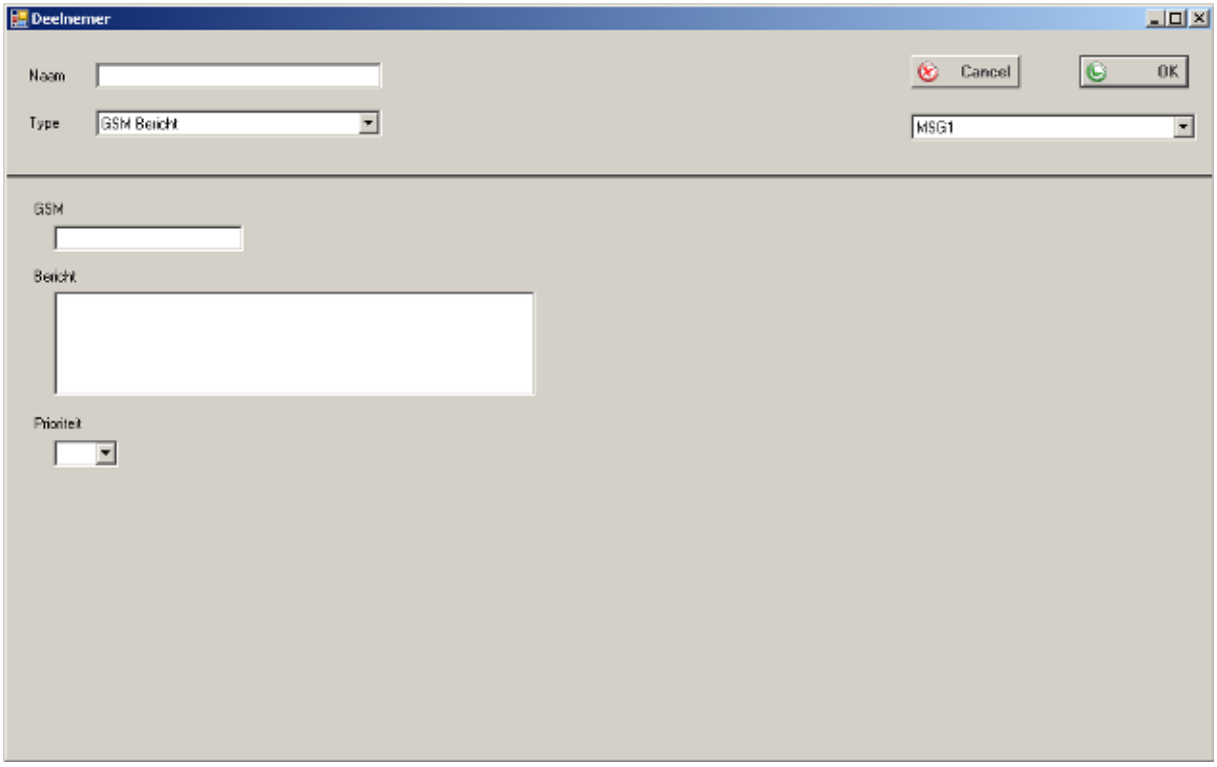
Group by TTL: Group that is called after TTL expires.

Re-trigger: This can be used to reset the repeat time.

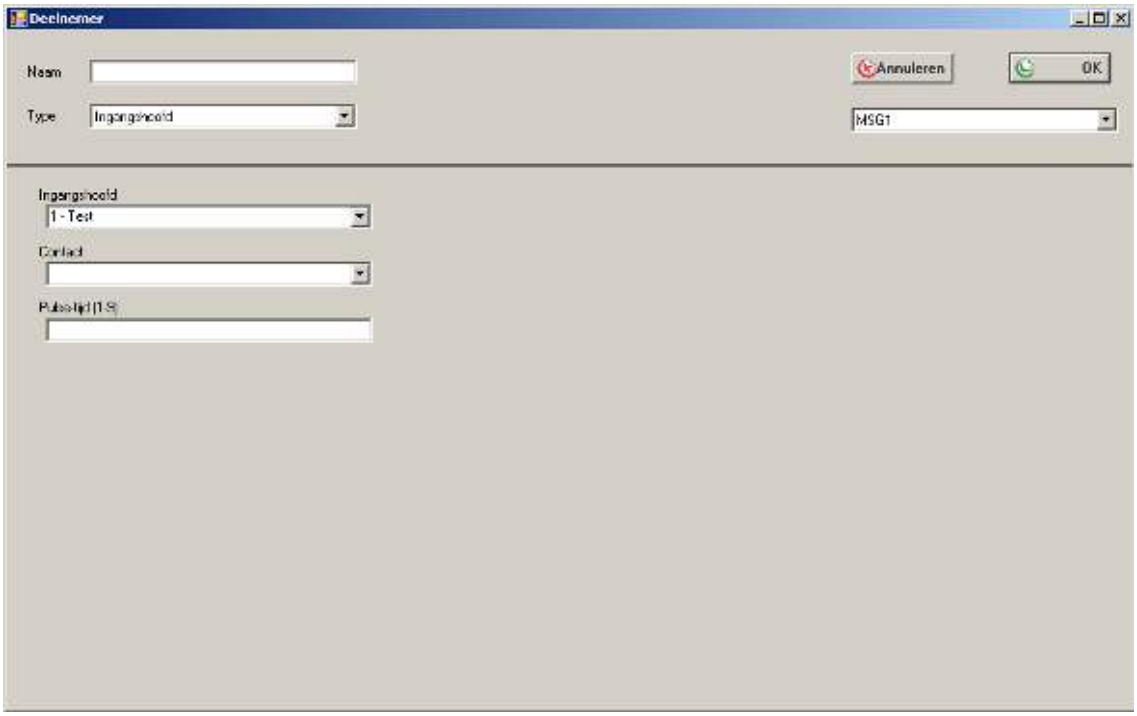
8.7 Repeat call reset



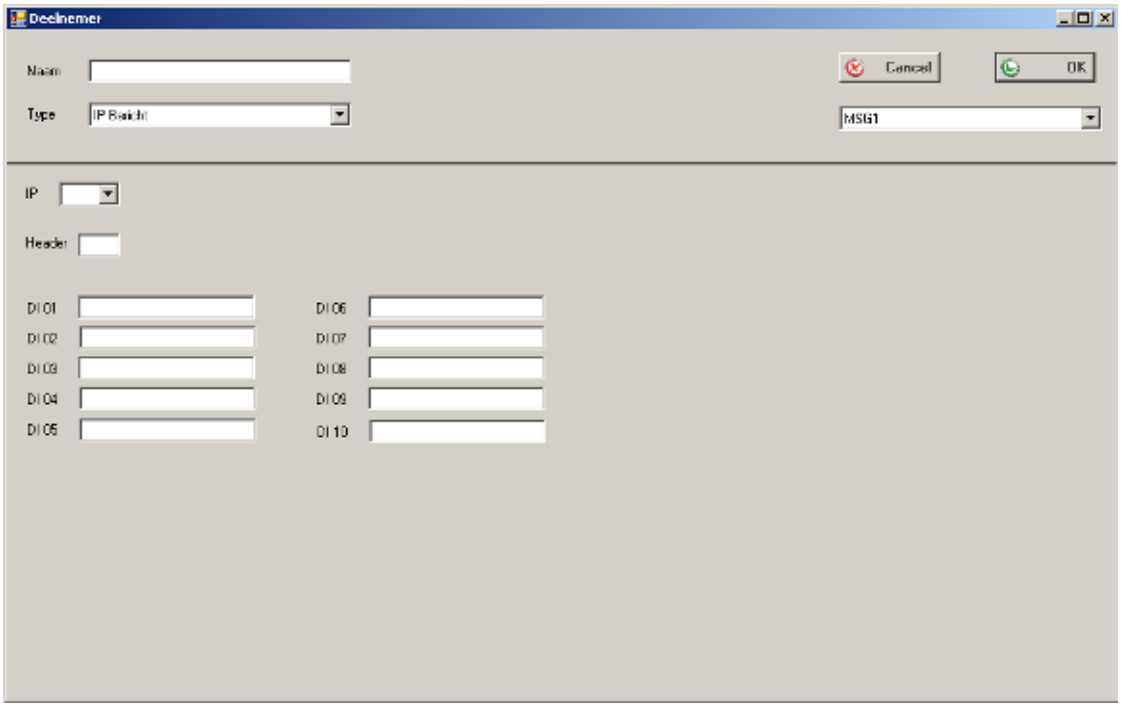
8.8 Mobile phone



8.9 Contact head

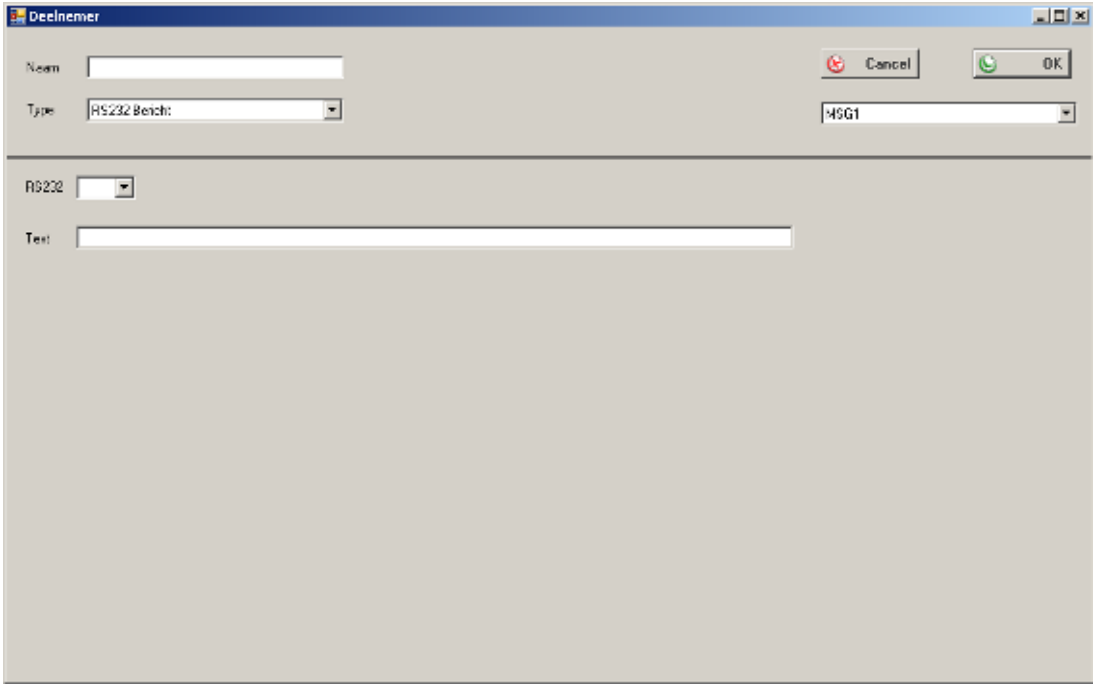


8.10 IP



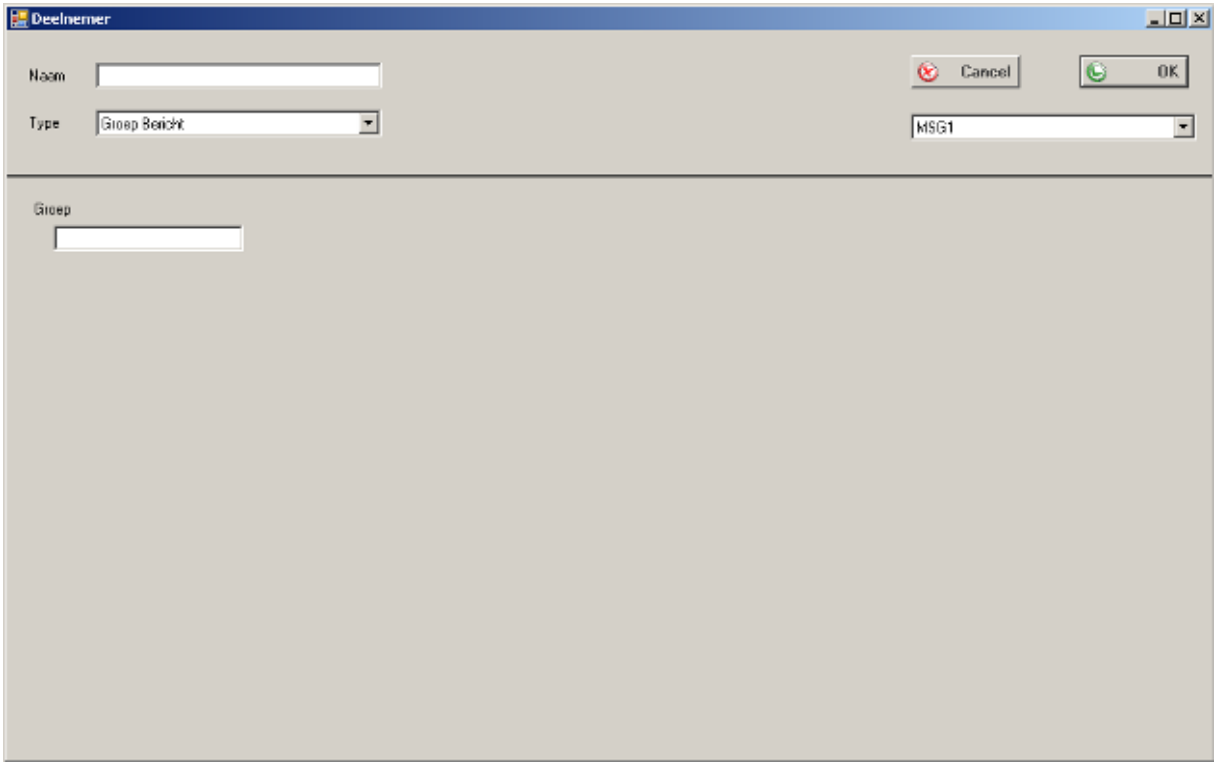
IP: ID of the interface
Header + ID = as described in the protocol

8.11 RS232

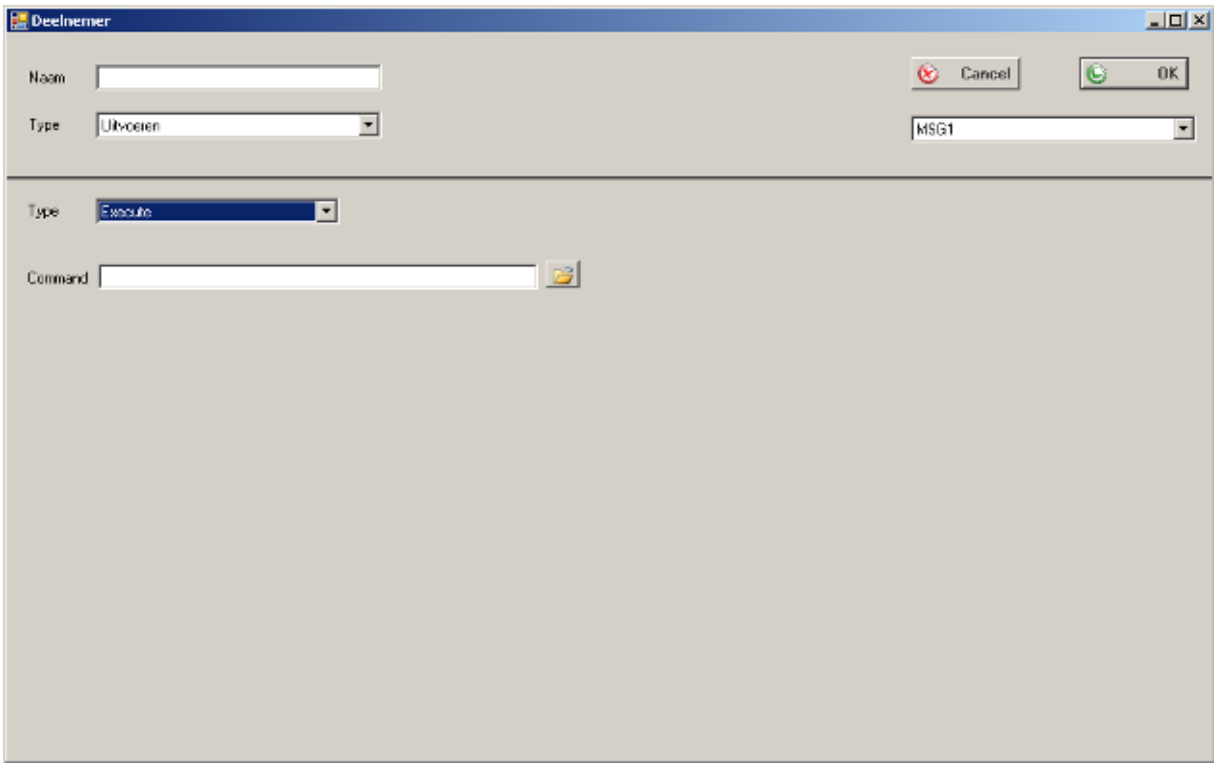


RS232: ID of the interface

8.12 Group



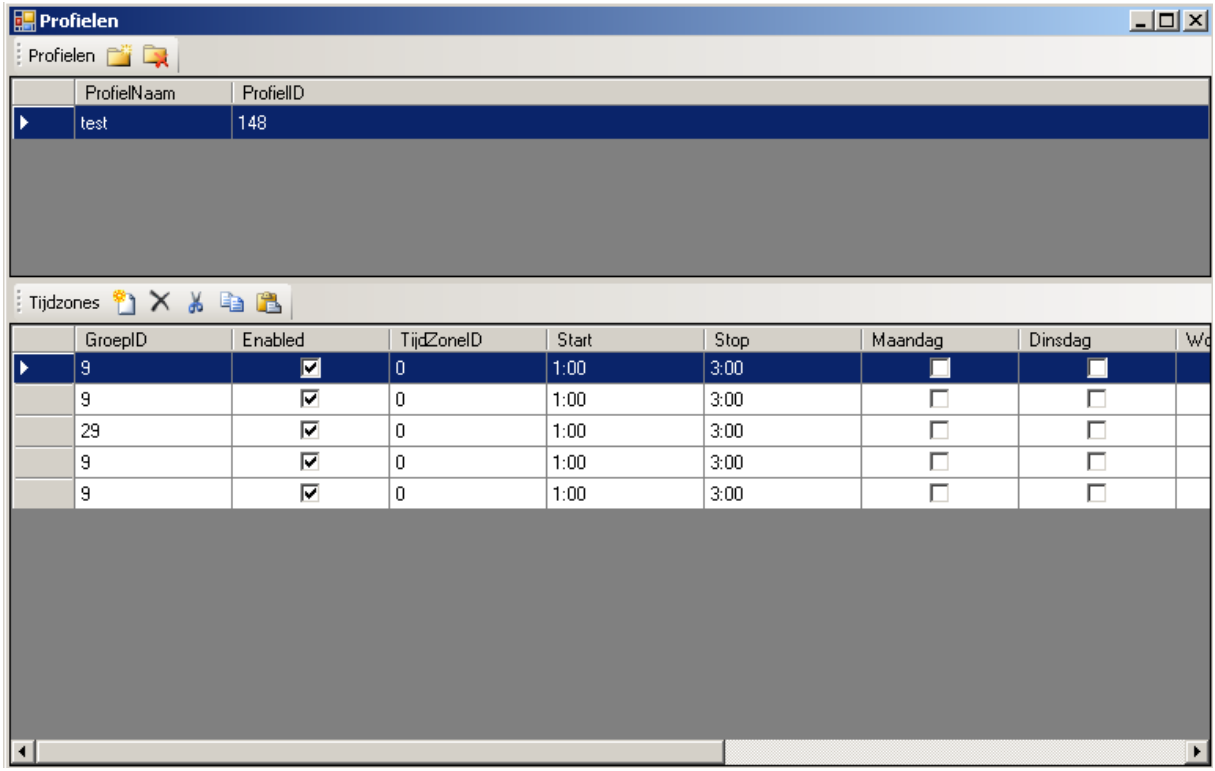
8.13 Execute



Command: Location of file to be executed

9. Profiles

A maximum of 10,000 profiles can be defined in the system, and each profile consists of an unlimited number of criteria. The criteria in the profile determine which group is active at a certain time on a certain day. Two groups can be active during a particular period.



10. Conversion tables

The software allows use of an unlimited number of conversion tables for converting certain information to a more useful format. These tables can be used for compiling the members of a group.

11. Arranging windows

The windows can be arranged in three different ways: horizontally, vertically or as a cascade.

12. Calling users

Every output model can be used to perform a test call from the main menu.

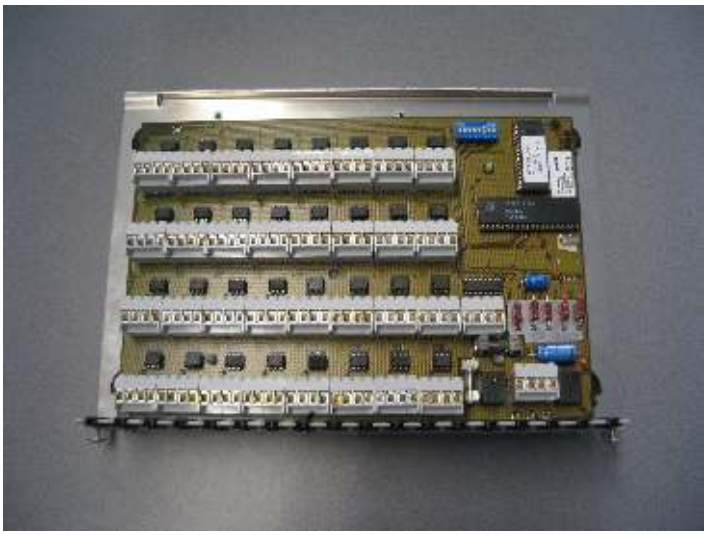
13. Backup/restore

The complete programming can be transferred from one server to another or copied to a backup via Backup and Restore. All the important files are compressed into a single ZIP file for transfer to the destination. All the data is kept in XML files, and the settings are in the MRS.ini file. When moving to another server, it is also necessary to obtain a new licence file.

14. Operation

14.1 Contact head

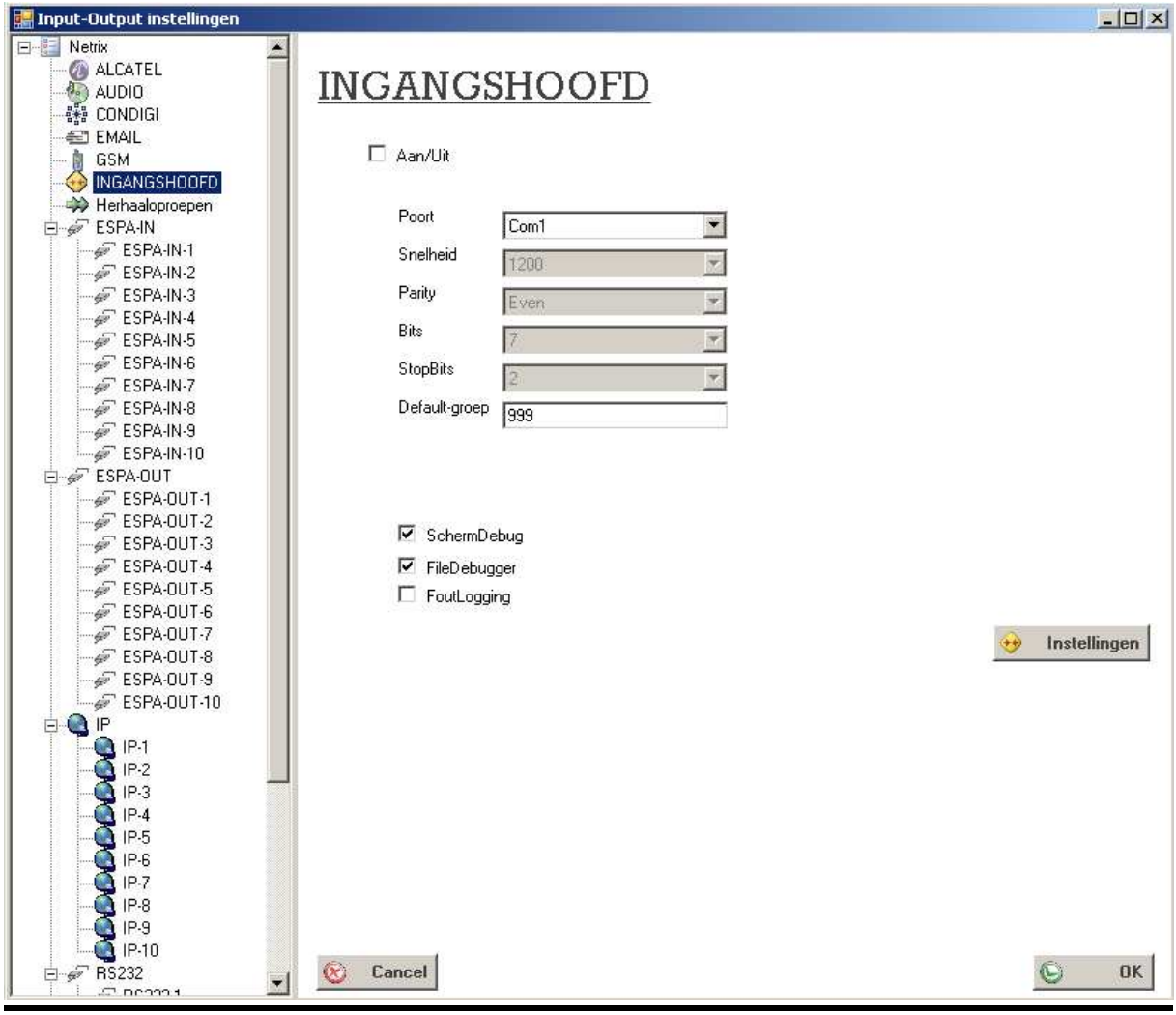
The contact head is a BOSCH product that has been in use for many years to accept input from contacts and send output to pagers.
Through use of an interface, we can connect these to a COM port of the Netrix server. Every change on a contact is passed on and can be processed within the Netrix. Multiple contact heads can be connected simultaneously.
Communication with the contact head is indicated by a green or red colour.
Starting with version 2.0.0, the two outputs provided on the contact head can also be used.



(two output contacts at bottom right)

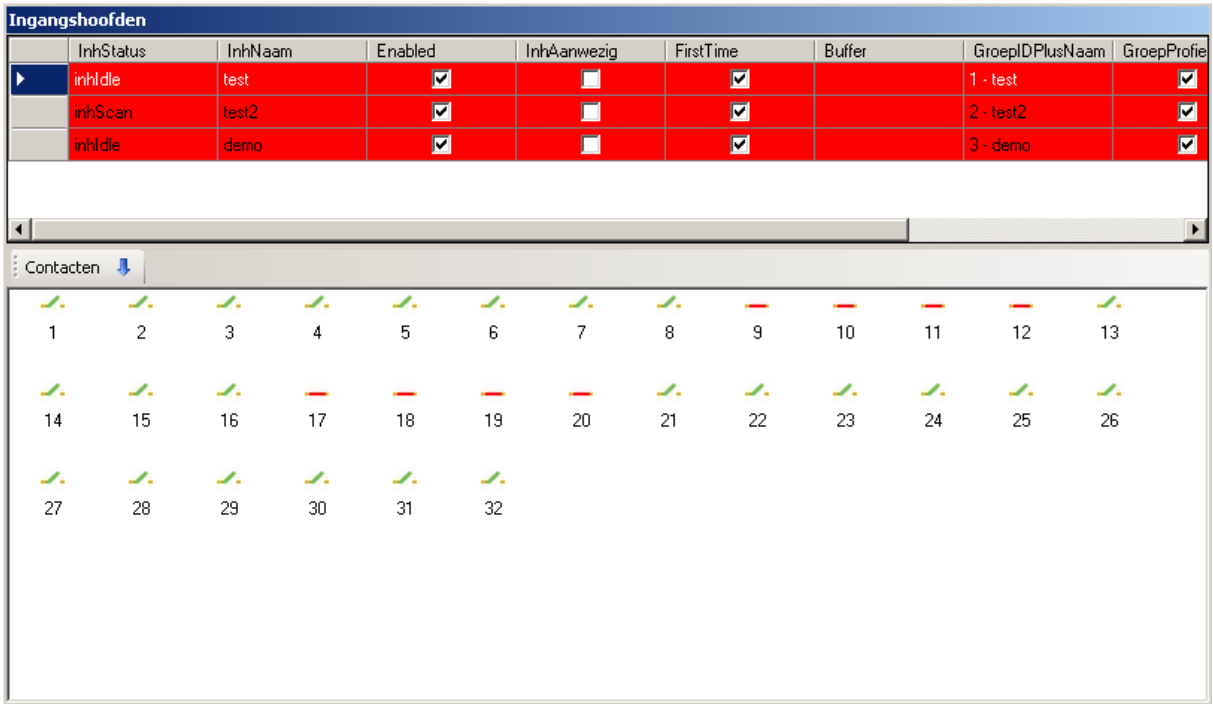
14.2 Adapter

14.3 Contact head settings



On/Off:	Switches on this input module
COM port:	Port number to which the contact head is connected
Speed:	The baud rate used for communication between the computer and the contact head interface (CANNOT BE CHANGED)
Bits:	The number of bits that must be sent to the contact head interface (CANNOT BE CHANGED)
Parity:	The parity that must be set (CANNOT BE CHANGED)
Stop Bits:	The number of stop bits that must be sent (CANNOT BE CHANGED)
Screen Debugger:	If this is active, all messages sent and received by the contact head are shown in a debug screen.
File Debugger:	If this is active, all messages sent and received by the contact head are saved in a file.

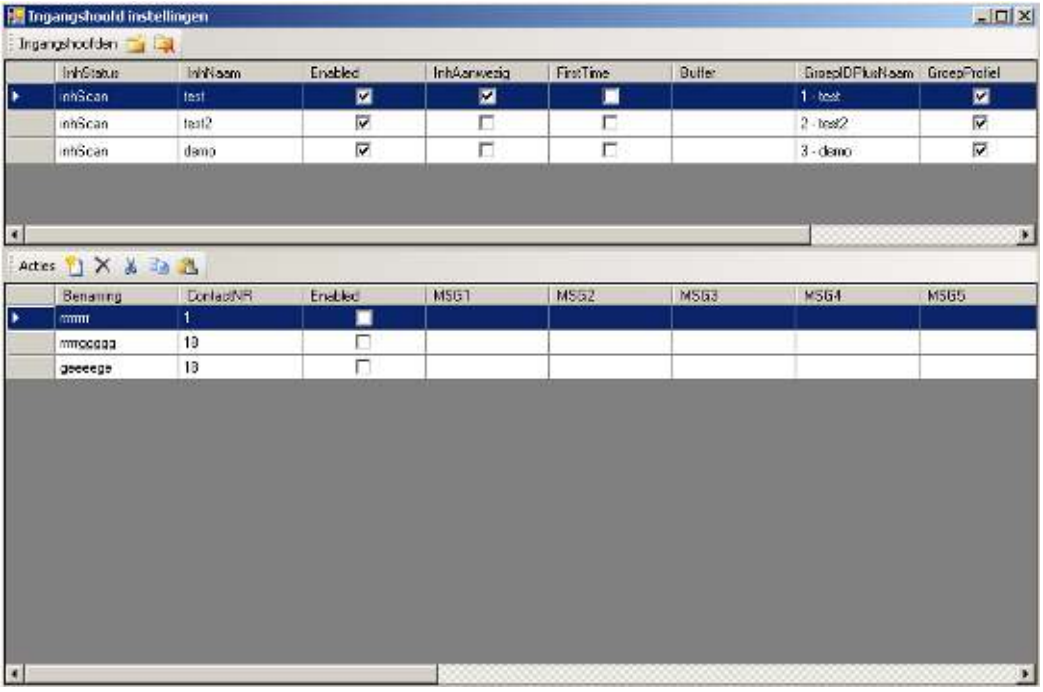
14.4 Contact head work screen



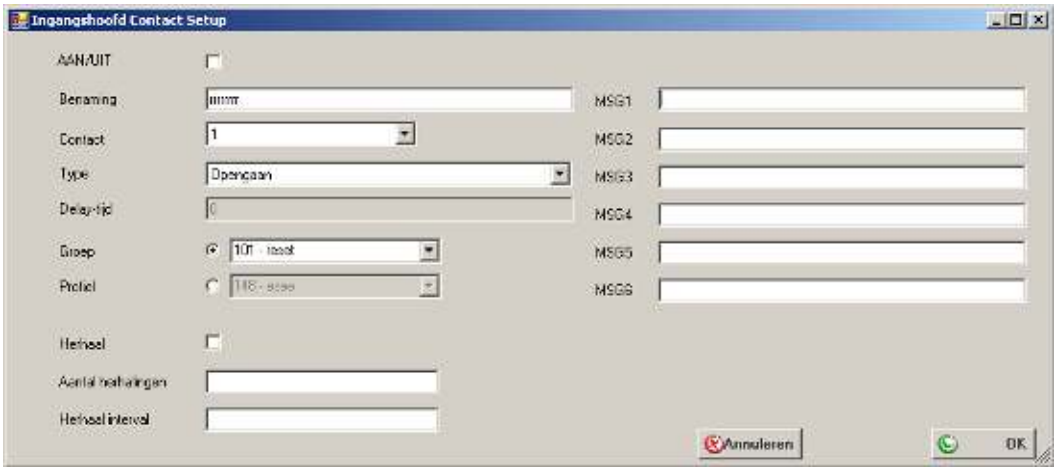
The contact heads, along with their serial numbers and names, are shown at the top of the screen. An overview of the contacts is shown at the bottom.

14.5 Contact head configuration

Every contact can be assigned an action, which can then place a call to a group. Valid contact types are: closing, opening, delayed closing and delayed opening.
Six variables can be passed on to the groups; these variables can be used how ever you like.
Variables: MSG1
MSG2
MSG3
MSG4
MSG5
MSG6



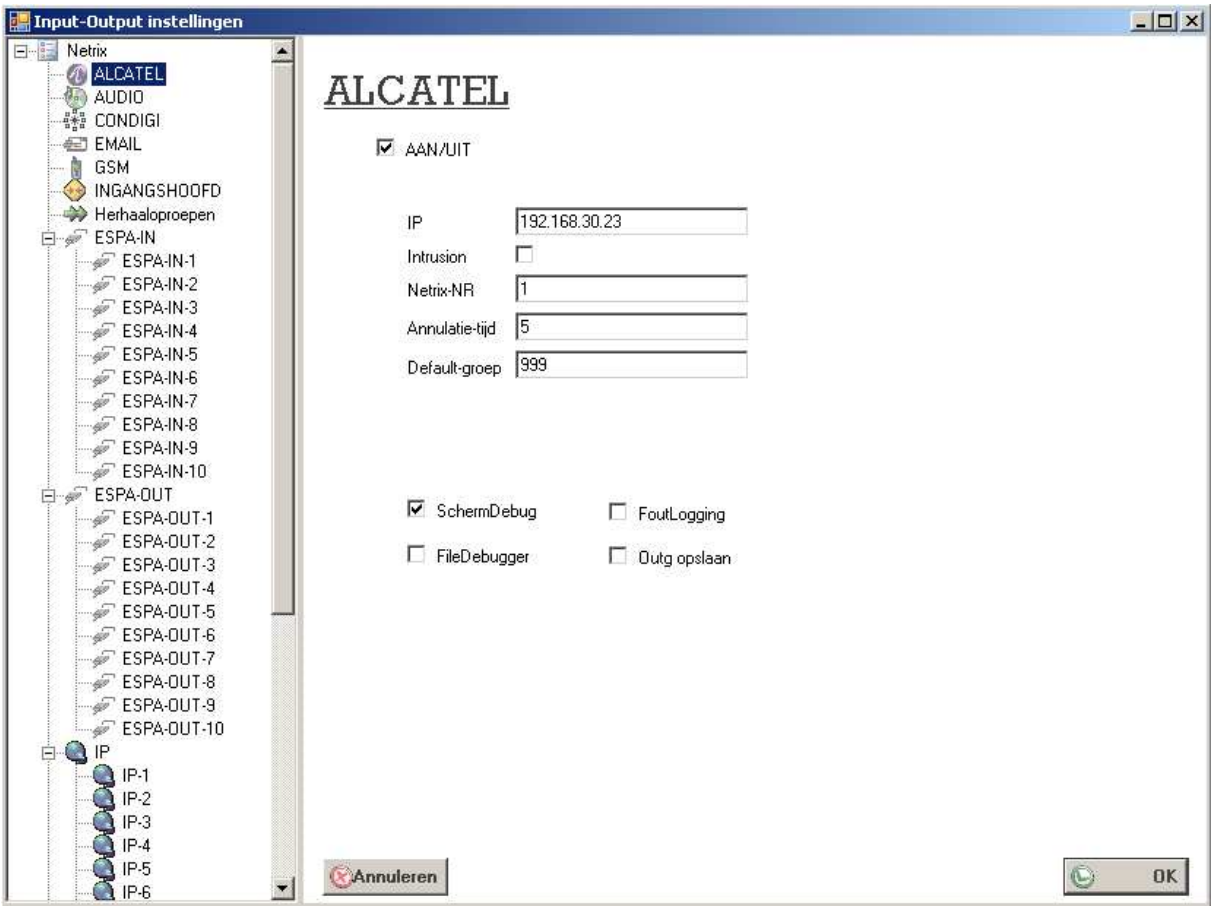
Name: Name of the contact head
Error Group: Group that must be called if there is a communication failure



ON/OFF:	Used to switch on a contact
Group:	The group that this contact must call
Profile:	The profile that this contact must call
MSG1:	Message variables
MSG2:	Message variables
MSG3:	Message variables
MSG4:	Message variables
MSG5:	Message variables
MSG6:	Message variables
Name:	Name of the contact
Type contact:	Opening, Closing, Delayed opening, Delayed closing, Pulse contact
Repeat:	Start repeat call for this group
Number of repetitions:	Number of times the call must be made
Repeat interval:	Time between the repeated calls in seconds

15. Alcatel

The Alcatel interface is used to send messages to any Alcatel telephone via an Alcatel telephone system (OXE).
The phones can be DECT, IP or Digital phones.



On/Off:	Used to switch on this output module
Intrusion:	Hands-free intrusion
Netrix number:	The internal phone number of the Netrix; this is required for intrusion.
Cancellation time:	The number of seconds that a phone rings when a message is received.
Default group:	A Netrix group that is called if the message cannot be delivered.
Screen Debugger:	If this is active, all messages sent and received by the interface are shown in a debug screen.
File Debugger:	If this is active, all messages sent and received by the interface are saved in a file.

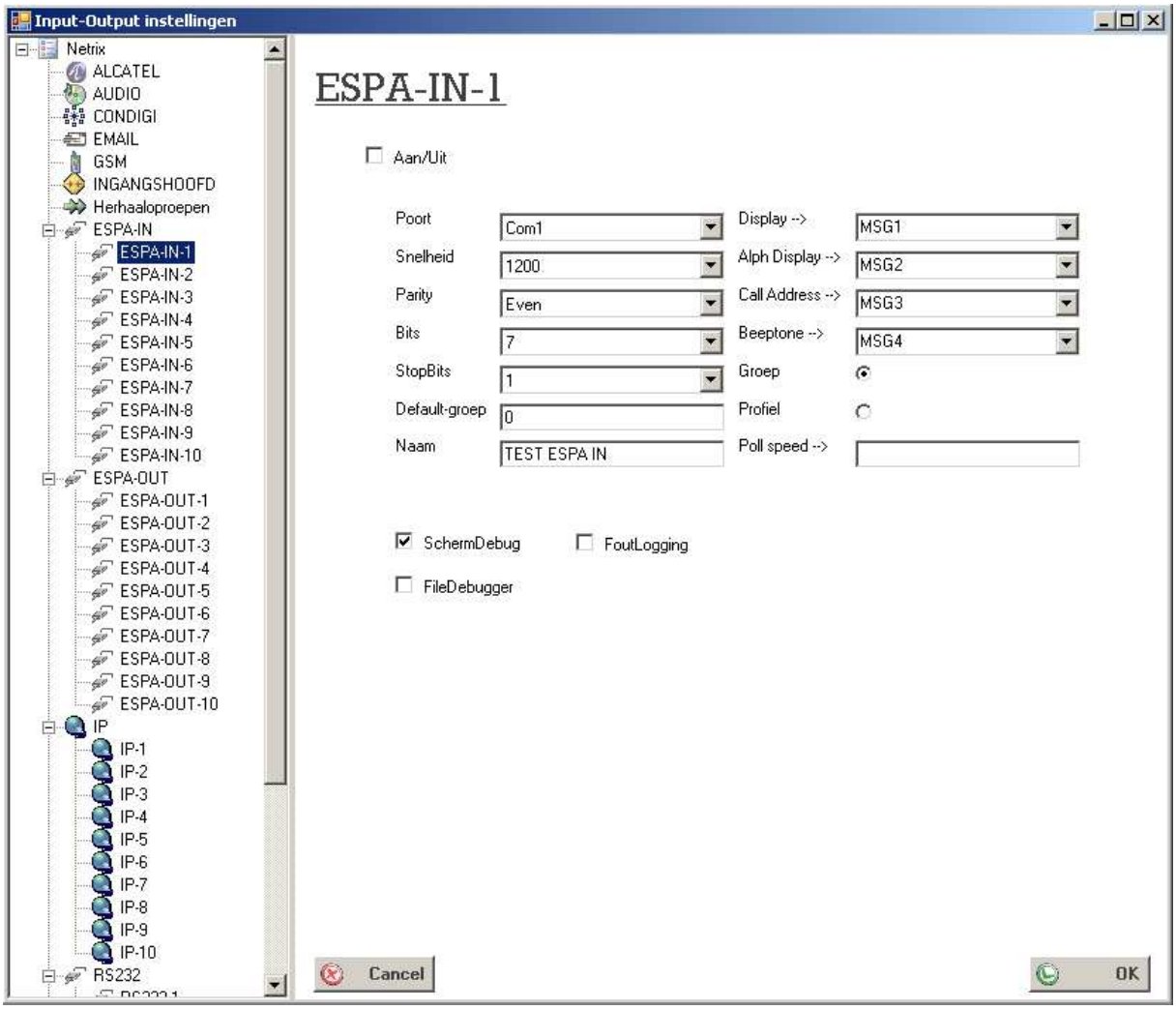
16. ESPA-IN

ESPA is a protocol developed for the health care market. Using this interface, calls from a health care system can be transferred to a nurse call system. This link is made using an RS232 connection. The ESPA inputs work according to a very simple standard and adapt themselves, as far as possible, to the connected equipment.
With the ESPA protocol, the message fields are predefined.

Fields:

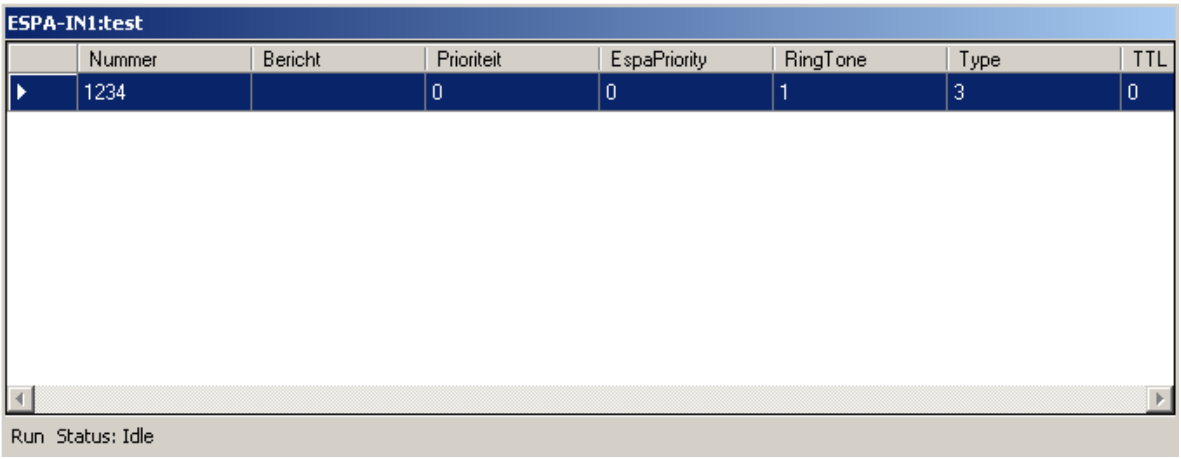
- Display information
- Call type
- Pager number
- Beep tone

Within our system, the display information is further subdivided into an alphanumeric and numeric display.
During configuration of the ESPA interface, we can select where the fields must go. The group or profile that is called is the pager number. If this is longer than four characters, the Netrix only looks at the last three digits.

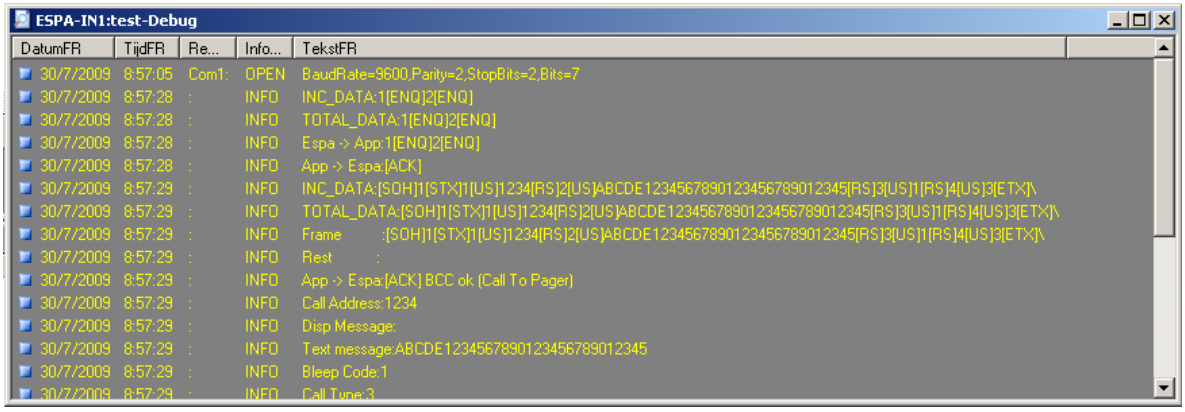


On/Off:	Switches on this input module
COM port:	Port number to which the ESPA interface is connected
Speed:	The baud rate used for communication between the computer and the ESPA interface
Bits:	The number of bits that must be sent to the ESPA interface
Parity:	The parity that must be set
Stop Bits:	The number of stop bits that must be sent
Screen Debugger:	If this is active, all messages sent and received by the ESPA interface are shown in a debug screen.
File Debugger:	If this is active, all messages sent and received by the ESPA interface are saved in a file. This file has the name 'espaXXPM.dbg', where XX is the day of the month. If the file is already present, it is overwritten. The system maintains the debug files for a maximum of one month. In order to retain multiple months, the key DebugOverWrite, which is normally set to True, can be changed in the file MRS.ini. If this key is set to False, the data will always be appended to the end of the debug file.
Display:	The variable in which the display field should be placed. (e.g. MSG1)
Alpha display:	The variable in which the alphanumeric display field should be placed. (e.g. MSG2)
Call address:	The variable in which the call address field should be placed. (e.g. MSG3) This is also the group or the profile number that is called.
Beep tone:	The variable in which the beep tone field should be placed.
Group/Profile:	Selects whether a group or a profile must be called
Polling speed:	This is the interval at which the Netrix polls the ESPA input. This is set to 500 ms by default.

16.1 ESPA control screen

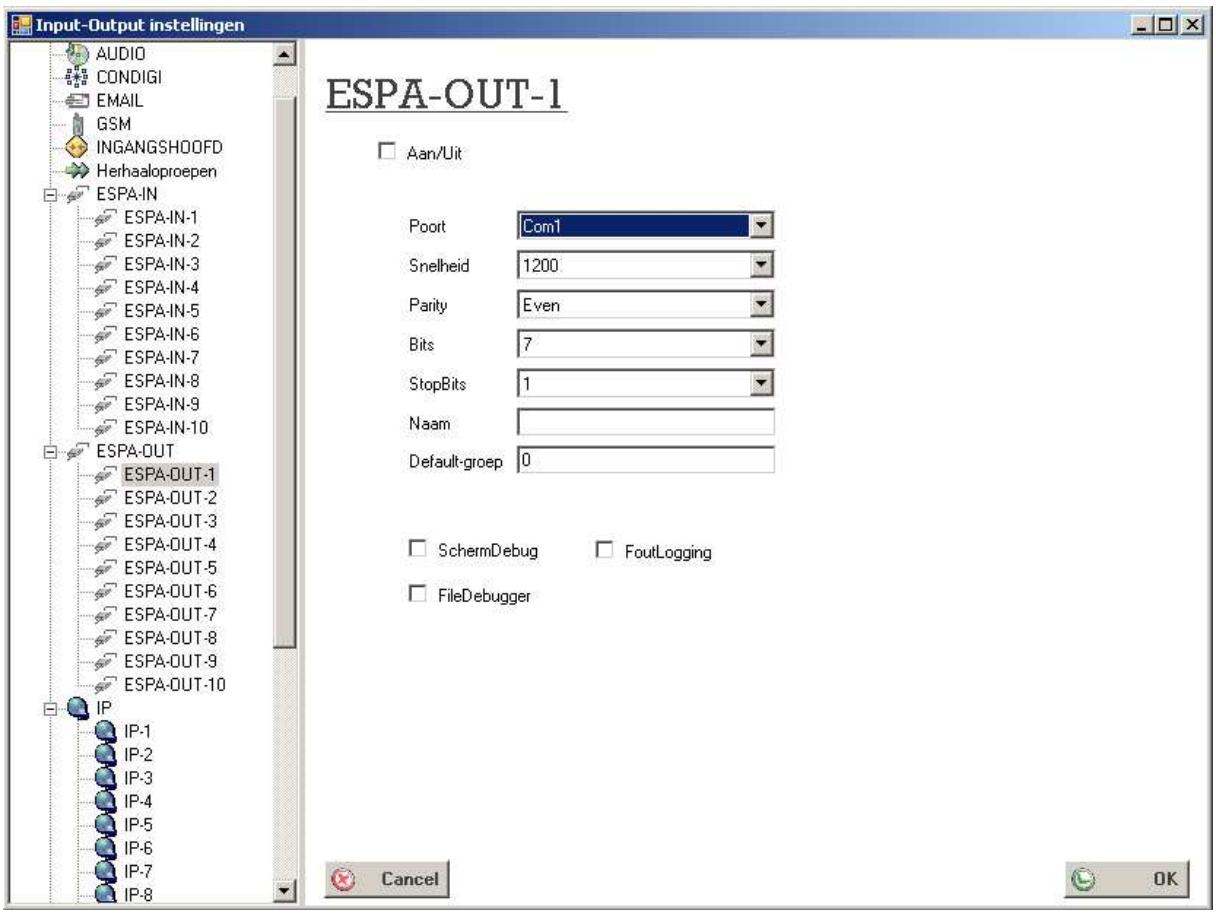


16.2 ESPA debug screen

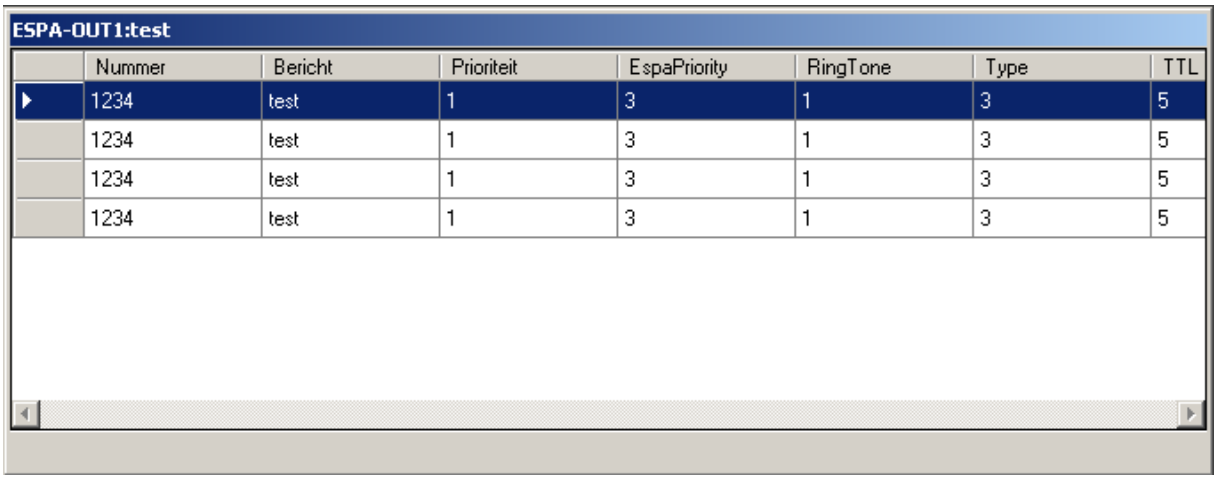


17 ESPA-OUT

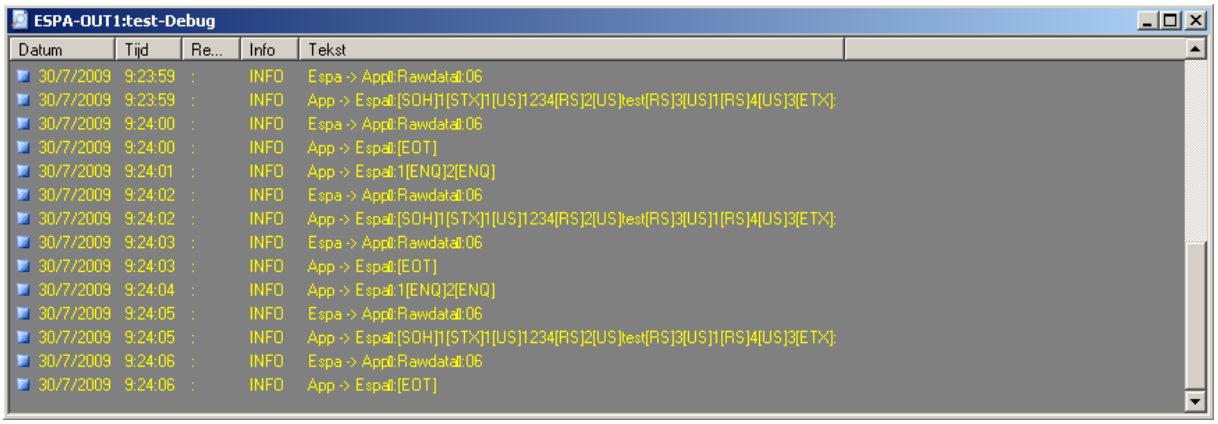
This module is used for sending ESPA messages, e.g. to telephone systems and for paging.



17.1 ESPA-OUT control screen



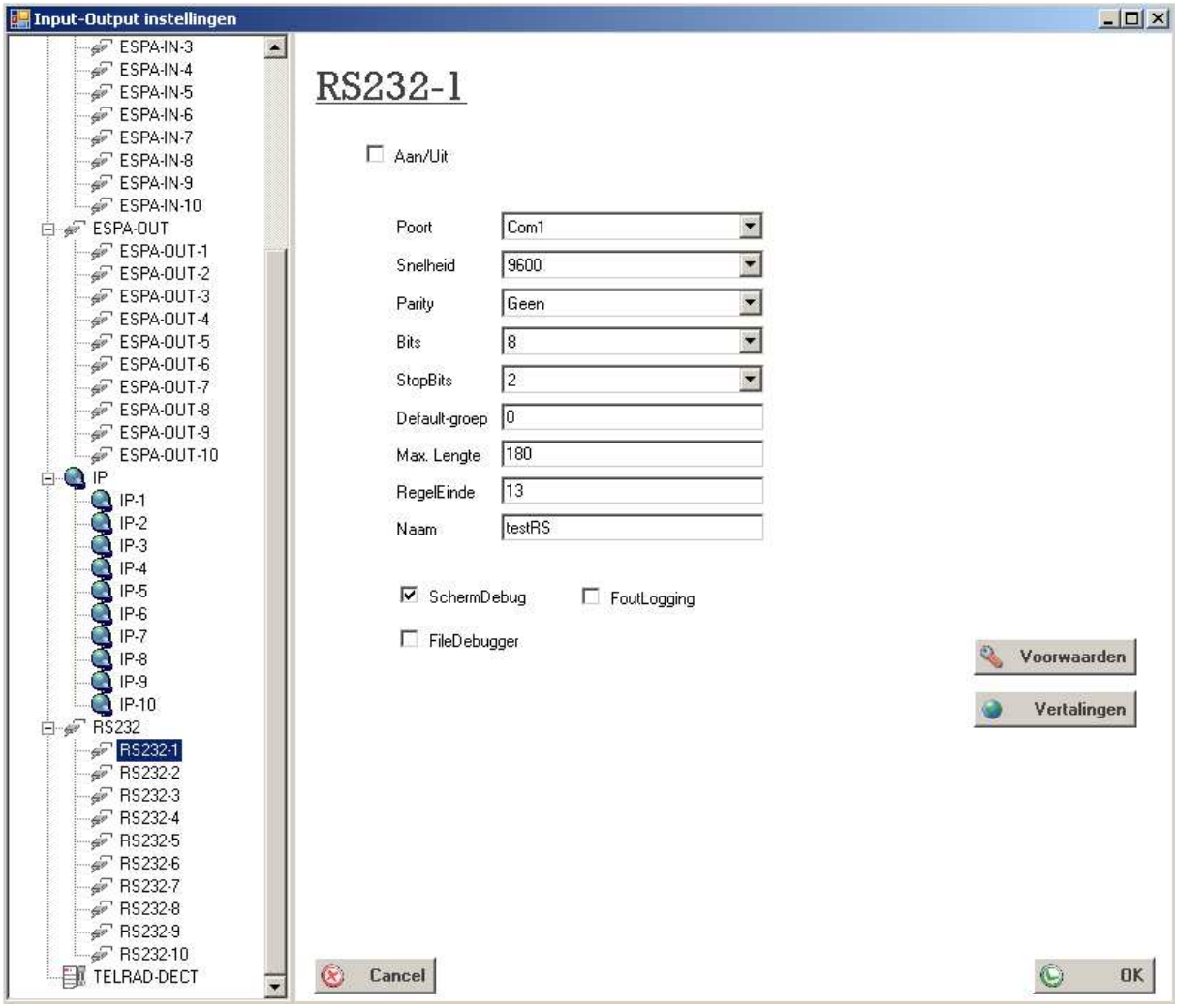
17.2 ESPA debug screen



18. RS232

From the RS232 screen we can link any RS232 output, analyse the data and take an action accordingly. Fire alarm consoles are often linked in this manner so Netrix can receive fire alarm notifications that specify the zone in which fire has been detected.
This information is then analysed and compared to conditions that are used to determine which group or profile should be called.

18.1 RS232 configuration

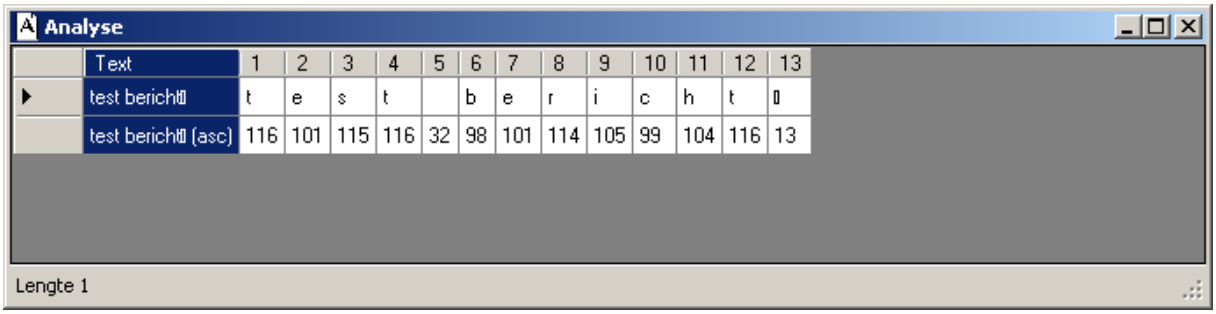


On/Off:	Switches on this input module
COM port:	Port number to which the RS232 interface is connected
Speed:	The baud rate used for communication between the computer and the RS232 interface
Bits:	The number of bits that must be sent to the RS232 interface
Parity:	The parity that must be set
Stop Bits:	The number of stop bits that must be sent
Screen Debugger:	If this is active, all messages sent and received by the RS232 interface are shown in a debug screen.
File Debugger:	If this is active, all messages sent and received by the RS232 interface are saved in a file.
Error Logging:	If this field is set, all messages that are not analysed are saved in a file.
Name:	This field contains a representative name.
End-of-line:	These are the characters used to mark the end of an RS232 message (ASCII separated by ';').

18.2 RS232 debug screen

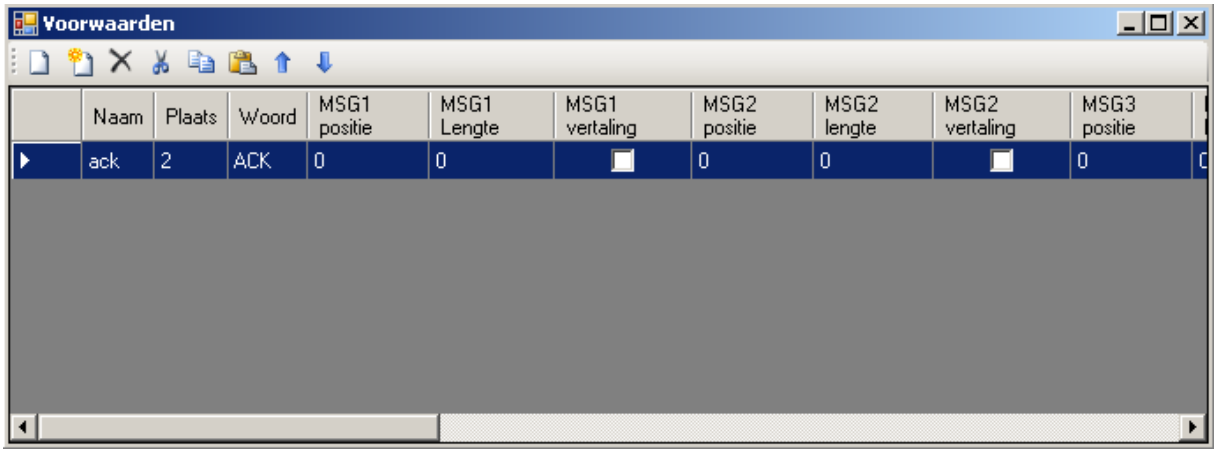
18.3 RS232 analysis screen

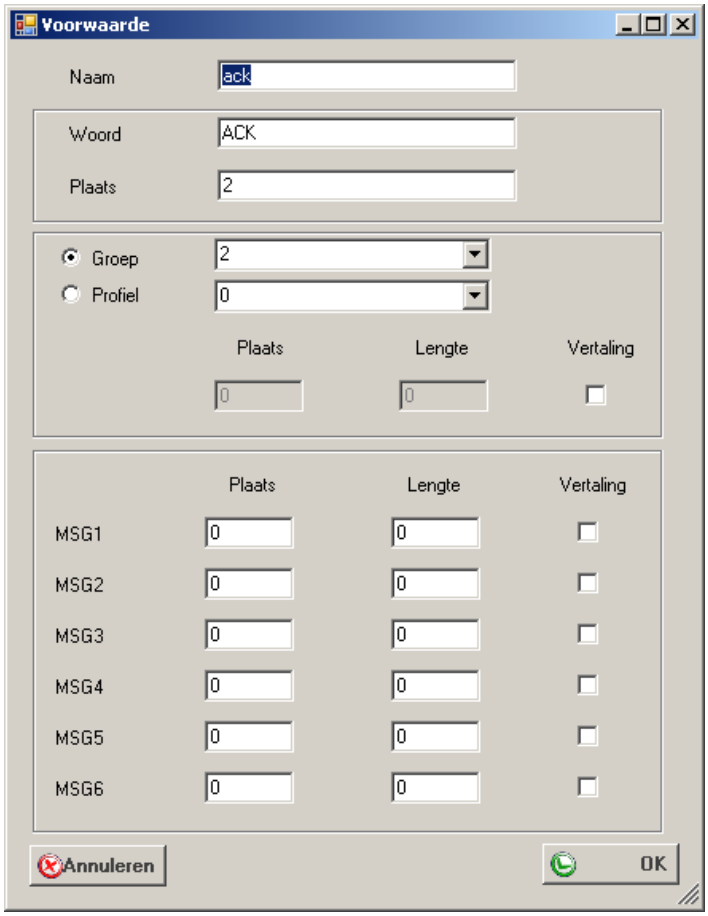
With this screen you can view the content of incoming information and use it to establish the conditions. This screen appears if you select a line in the debug screen and press the 'a' key.



18.4 RS232 condition rules

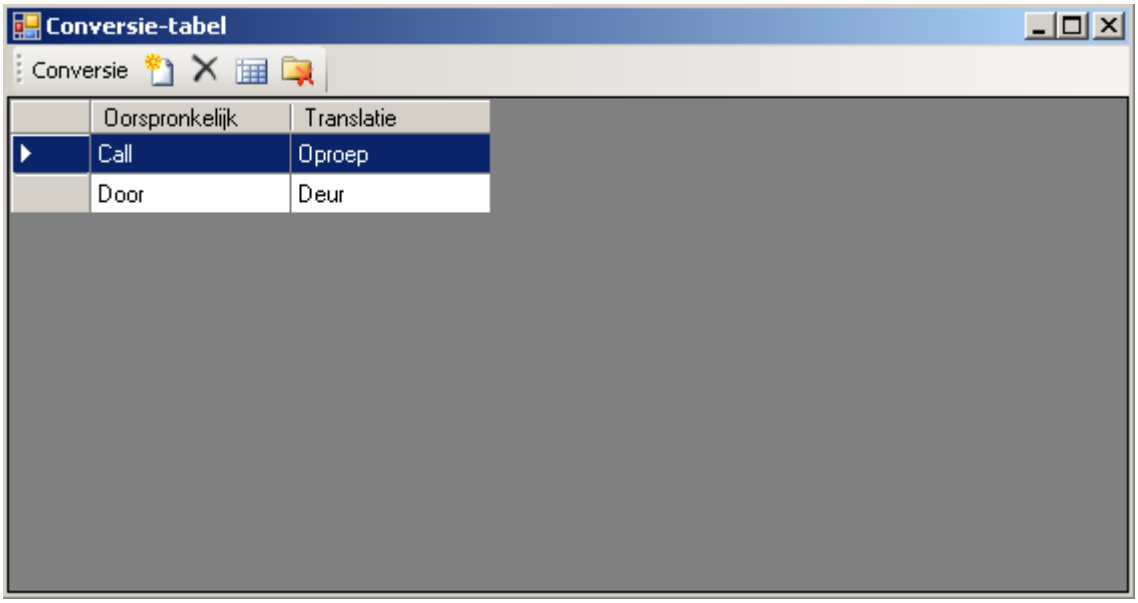
These rules are used to determine what is done with the incoming data. If one of the conditions is met, this rule will be processed according to the rules of this condition.
The condition consists of a word that must occur at a certain position in the rule; once found, all the variables are collected from the rule at specified positions, for a specified length.





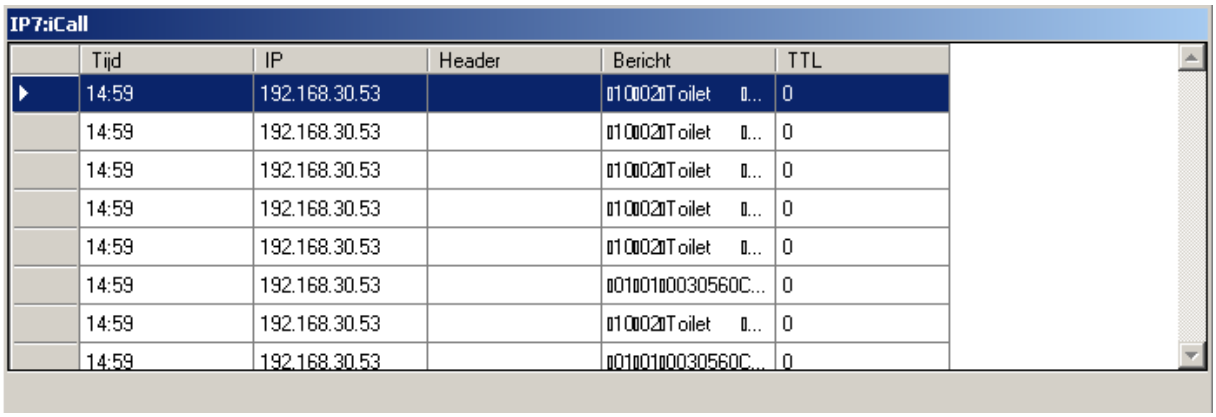
18.5 RS232 conversion rules

The filtered information can then be compared to a list for conversion. If this word does not occur in the conversion list, the variable retains its original value.



19. IP

This module makes it possible to read and send TCP/IP information. Both TCP and UDP are supported. This module is mainly used to read in information from the iCall nurse call system.

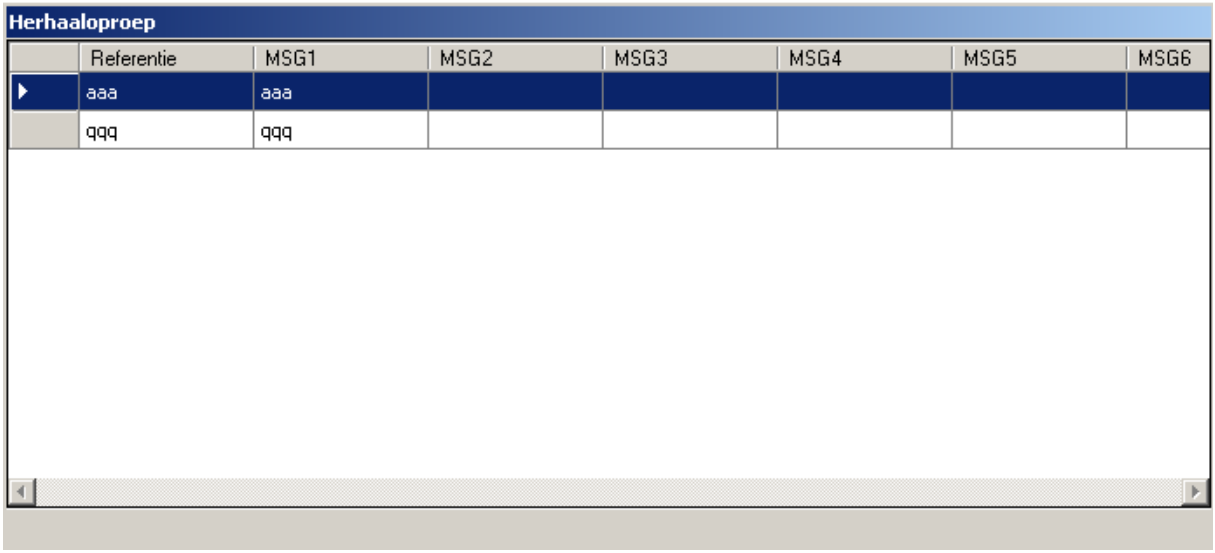


20. Repeat calls

This module makes it possible to start a repeat call. A reference is included in the list, and this reference must also be used to stop the call.
The interval between calls can be set to a length of up to 5 minutes.
The repeat calls can be assigned a different colour to make it easier to identify the repeat calls.

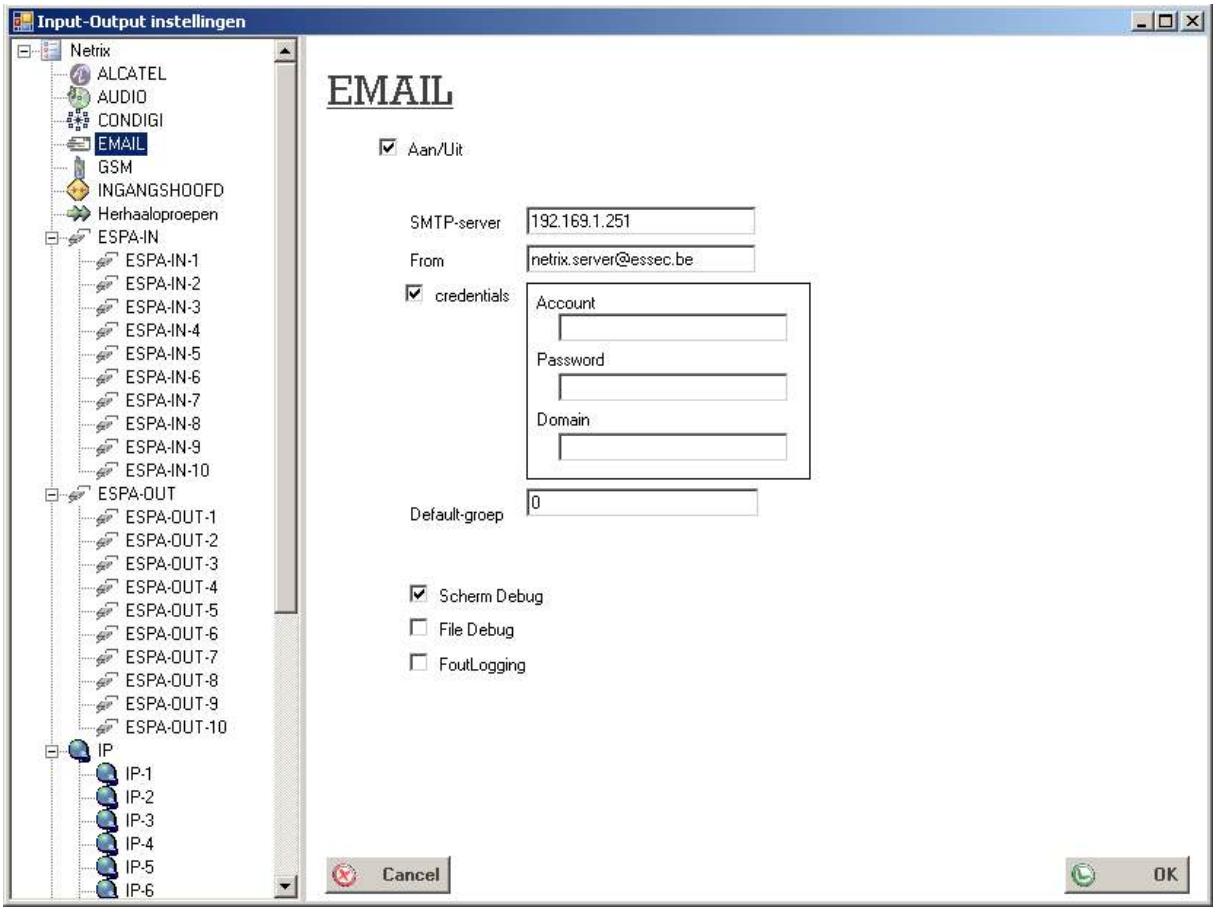
Note: This interface does not have a debug screen.

TTL=0 The last call only goes out to the TTL group
TTL=1 The last call only goes to the called group and the TTL group



21. Mail

This module makes it possible to send e-mail messages. Optionally, one attachment can be sent with each e-mail.



22. Condigi

This interface makes it possible to register calls from wireless Condigi pendants. The calls are sent to the Netrix via a Condigi Wireless Receiver IP. The receivers filter out double calls themselves (interval between calls can be set in the website of the receiver), and the Netrix does this too (in the case of calls coming from two different receivers).

Various call groups can be assigned to each pendant. This is necessary so the calls can later be sent to the right group.
If 'automatic learning' is activated, all Condigi pendants that place calls are configured with a standard set of groups. Various groups can be set for the following types of calls:

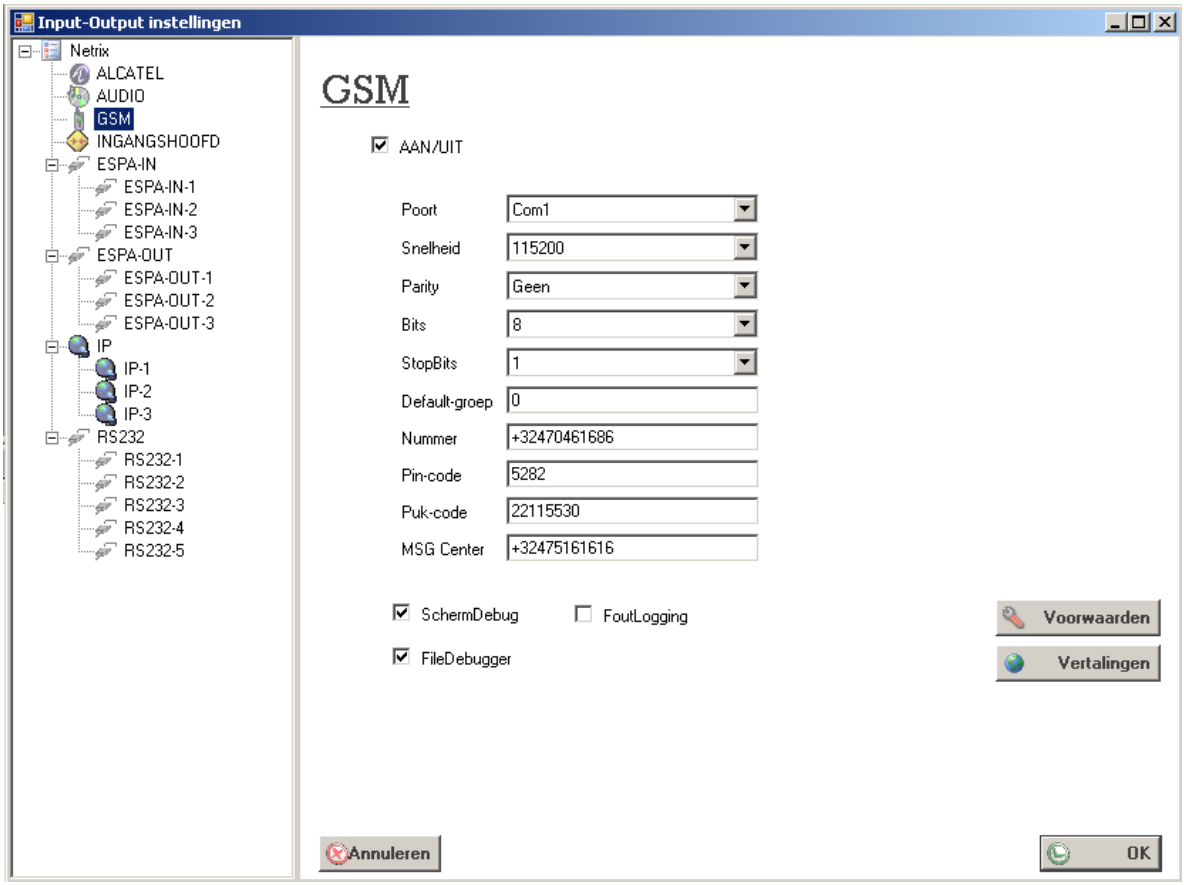
- Call
- Call type2
- 3Push
- Battery-low
- New Position

Every pendant can also be temporarily deactivated.

To obtain an overview of the various receivers, use the menu option Info -> Condigi.

23. Mobile phone

The mobile phone interface is linked to a mobile phone modem (Wavecom, Siemens). This module takes care of configuring this modem properly. Some information is also retrieved from the modem, such as the signal strength and quality. The software can make use of SIM cards with or without a PIN code. The signal strength and quality are shown at the bottom left of the interface.
The connection between the modem and the PC is made via a standard RS232 cable.

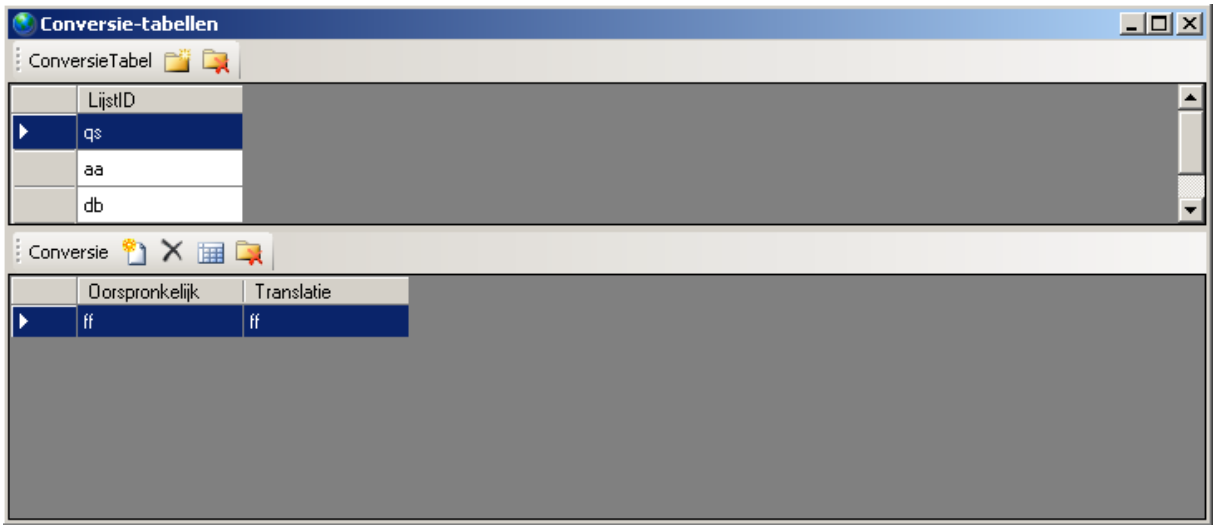


24. Conversion tables

Just as with RS232 or IP, we can add additional conversion tables.
To do so, use the menu option Setup -> Conversion tables.

First create a new conversion table, and then add your conversions.
You can also easily import a CSV file, which is handy if you need to add many conversions.

Note: You will need the name of the list later on if you want to use a conversion table to convert a variable in the groups.



25. Debugging

Screen debugging can be switched on for nearly all the interfaces. This is an additional window that opens, which provides you with more detailed information.

Note:
Screen debuggers can slow down the system.

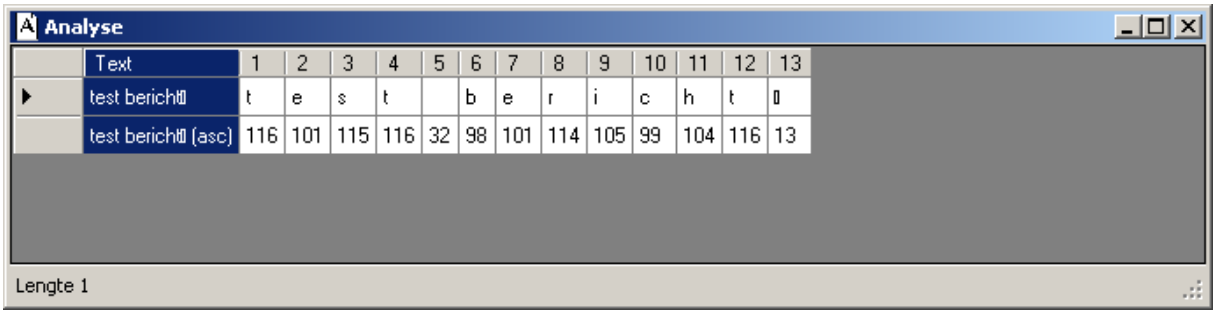
26. FileDebug

If FileDebug is on, all the information that is normally displayed in screen debug will be logged to a file.

Note:
File debuggers can slow down the system.

27. Analysis

If you press the letter ‘a’ in a debug window, you can analyse the information in the selected line.



28. Languages

Languages can be added easily.
Translate one of the .lng files to the desired language. Words between [square brackets] must not be translated.
Place this file in the root directory of the Netrix software and you will then be able to select this language using the menu option Options -> Language.

29. Conversions

Replace old Netrix software with new Netrix software.

Make a backup of the old Netrix files and Restore them in the new software.
Make sure a new licence file is available and copy it to the Netrix installation folder.

Restart the Netrix software.
Because DAT files were used in the old software, these files will be automatically converted to XML. The necessary modifications will also be made to the MRS.ini file.