



How to cascade/repeat SMS message and stop with a SMS reply.

If you wish to have SMS sent multiple times until one of the recipients acknowledges by sending a response, please follow this guide where we use groups, memory variable, timers, and incoming messages.

1. Set up a group of recipients in the USER section

These recipients will all receive the alarm messages when this group name is used in RULES:

The screenshot shows the Avior Suite 1.2.0.0 interface. The title bar indicates the file path is D:\SMS Cascade with Acknowledge.map*. The interface is divided into several sections:

- Toolbar:** Contains icons for file operations (copy, paste, delete, etc.) and a shield icon with a gear, which is circled in orange.
- Users List:** A table on the left with columns for ID and Name. The first three rows are: 001 Dick, 002 Harry, and 003 Tom. The rest of the rows are empty.
- User Form:** A detailed form on the right for editing a user. The 'NAME' field is 'Dick'. The 'IDENTIFIER' field is '+447222222222'. The 'GROUP' field is 'service', which is circled in orange. The 'TYPE' dropdown is set to 'TELEPHONE', also circled in orange. Other fields include 'START DATE/TIME' (01/01/2000 00:00), 'STOP DATE/TIME' (31/12/2099 23:59), 'WEEKDAYS' (YYYYYY), and 'TICKETS' (UNLIMITED).
- Help/Info:** A section at the bottom right providing details about the 'USER NAME', 'USER TYPE', and 'IDENTIFIER' fields.

For each recipient enter their name and mobile number (+44...) and enter the group name you wish to use. In the above example we have three users, and they are all part of the same group named “service”

2. Setting up the first RULE to send a SMS when digital input 1 closes:

The screenshot shows the Avior Suite 1.2.0.0 interface. On the left, a list of rules is displayed, with '001 FirstSMSAlarm' selected. The main panel shows the configuration for this rule:

- NAME:** FirstSMSAlarm
- TRIGGER:** Digital Input 1 (din1)
- CONDITIONS:**
 - DIGITAL INPUT 1 - Value = 1
 - (Empty)
 - (Empty)
 - (Empty)
 - (Empty)
- ACTIONS:**
 - AT+SMS=service,"Alarm Message Text 1 - reply with OK to stop repeats"
 - mem1=1,300
 - mem2=1
 - (Empty)
 - (Empty)

Below the configuration, there is explanatory text:

For each event associated with the resources (channels) of the device, it is possible to define one or more behavioral rules, for a maximum of 500 total rules, freely divided between all possible TRIGGER events. Scheduled at TIME [hh: mm] event allows the use of wildcards.

Each rule is assigned a number that also represents the file name.

Within the rule are defined up to 5 optional CONDITIONS to be verified and up to 5 COMMANDS to be executed if the conditions are verified.

CONDITION
THIS IS ONE OF FIVE OPTIONAL CONDITIONS TO BE EVALUATED WHEN EVENT OCCURS

The RULE is giving a name “FirstSMSAlarm” and is using the

⚡ TRIGGER Digital Input 1 and when the

+ CONDITION Digital Input 1 = 1 (when the input closes) the RULE performs the following 3 ACTIONS:

> **AT+SMS=service,"Alarm Message Text 1 - reply with OK to stop repeats"**

(this sends an SMS to all the recipients identified in the GROUP names “service” and sends the message text “Alarm Message Text 1 - reply with OK to stop repeats”)

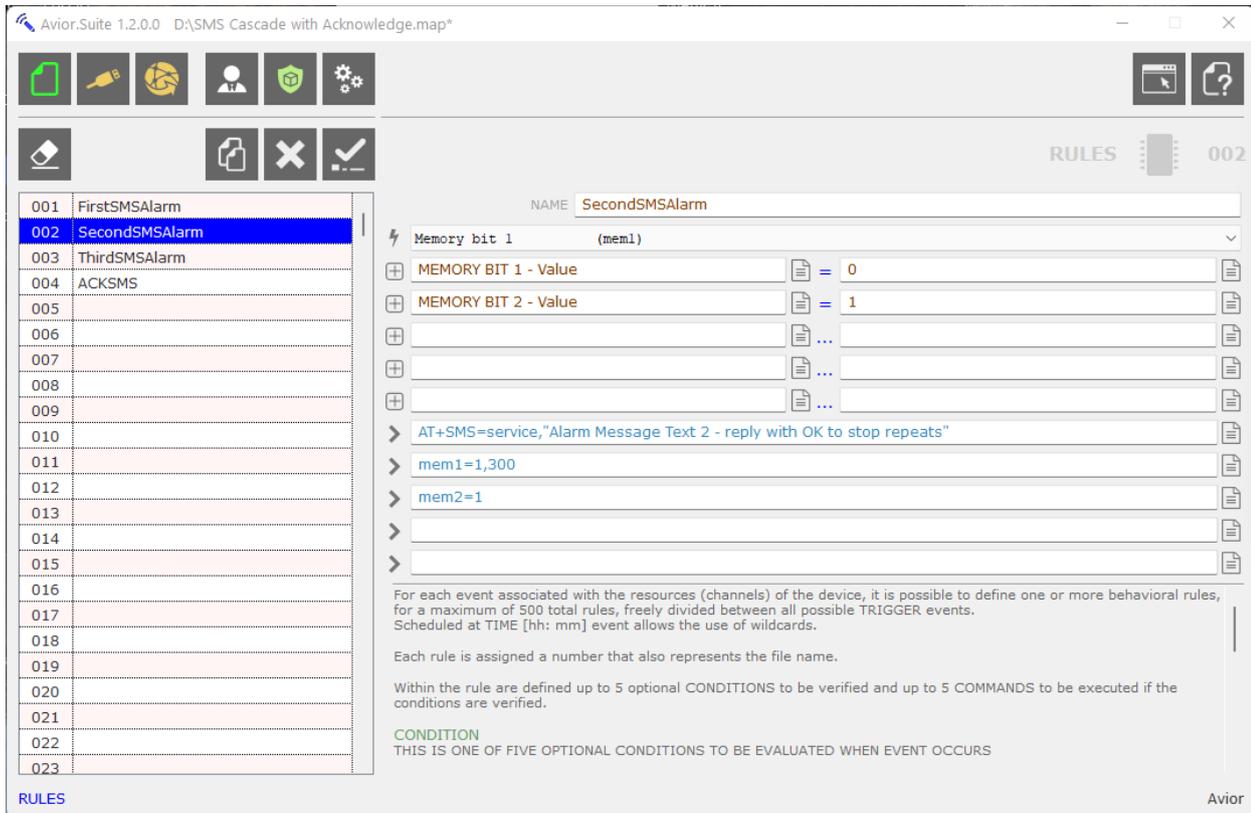
> **mem1=1,300**

(this turns internal memory bit1 ON for the time shown in seconds (300s = 5mins) and we will use this bit as a condition in the next rule – sending the message a second time)

> **mem2=1**

(this turns internal memory bit2 ON and is a condition that is controlled by RULE4 which is waiting for a user to send a reply by SMS to stop the messages being repeated)

3. The second and third RULES are almost the same – you could repeat these rules as many times as you wish if you want the message to be sent more than three times:



This RULE named “SecondSMSAlarm” is checking the condition of the two memory bits we set in the first RULE which was triggered by Input 1 closing. This RULE is TIGGERED:

⚡ TRIGGER Memory Bit 1 and when the

+ CONDITION Memory bit 1=0 (the 300 seconds timer in RULE1 has elapsed) AND a second condition Memory bit 2=1 (the incoming SMS alarm acknowledgment message has not be received) THEN perform the following 3 ACTIONS:

> **AT+SMS=service, \"Alarm Message Text 2 - reply with OK to stop repeats\"**

(this sends an SMS to all the recipients identified in the GROUP names “service” and sends the message text “Alarm Message Text 2 - reply with OK to stop repeats”)

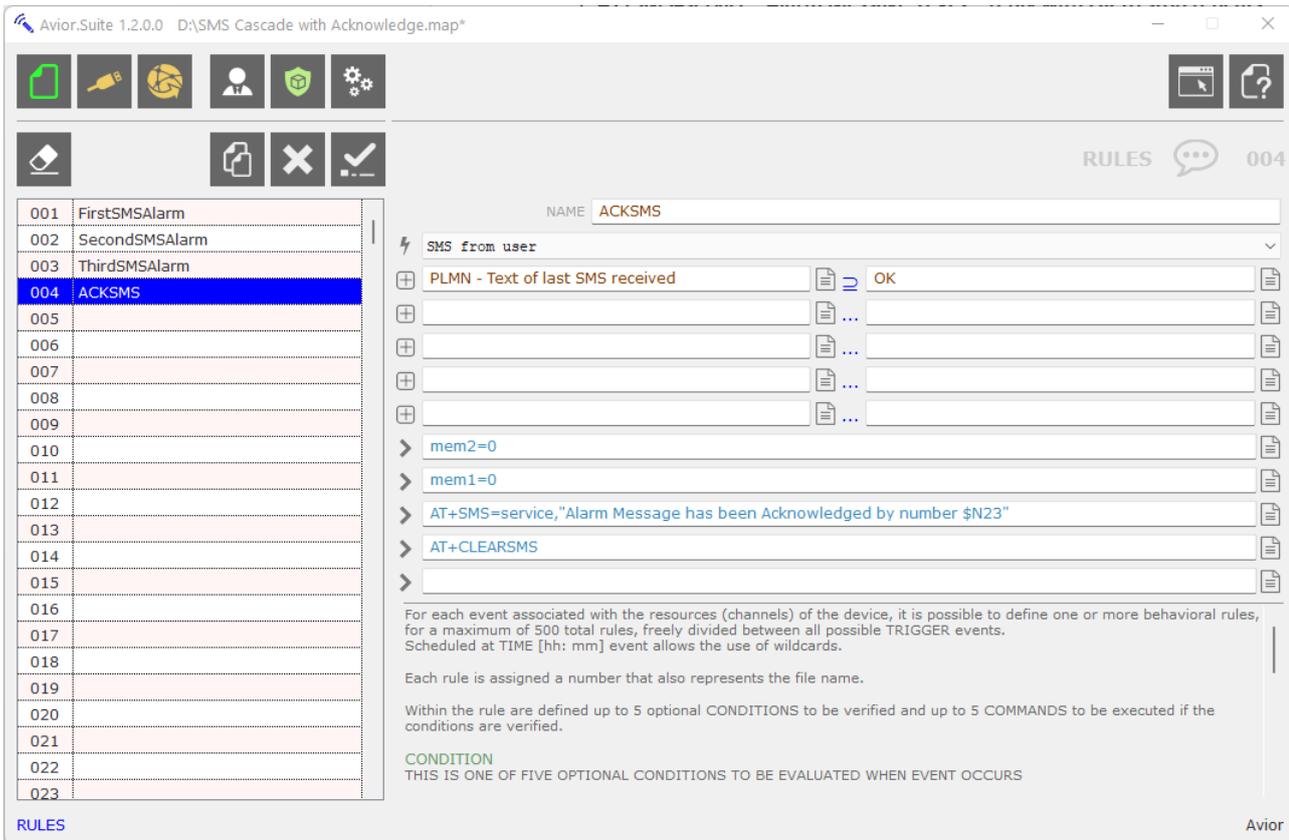
> **mem1=1,300**

(this turns internal memory bit1 ON, again, for the time shown in seconds (300s = 5mins) and we will use this bit as a condition in the next rule – sending the message a second time)

> **mem2=1**

(this turns internal memory bit2 ON and is a condition that is controlled by RULE4 which is waiting for a user to send a reply by SMS to stop the messages being repeated)

4. The fourth RULE checks for an incoming SMS message to stop the messages repeating:



This RULE named “ACKSMS” is checking for an incoming SMS from any USER containing the text “OK” The RULE is TIGGERED:

⚡ TRIGGER SMS from USER

+ **CONDITION** PLMN (mobile network) Text of the Last SMS received contains the characters “OK” THEN perform the following 3 ACTIONS:

> **mem2=0**

(this turns internal memory bit2 OFF and is a condition that used in the previous RULES as a CONDITION which stops the RULE and stops the alarm SMS being sent)

> **mem1=0**

(this turns internal memory bit1 OFF to rest any timers)

> **AT+SMS=service, "Alarm Message has been Acknowledged by number \$N23"**

(this sends an SMS to all the recipients identified in the GROUP names “service” confirming that the alarm has been acknowledged and inserts the number of the person which sent the OK message \$N23)

> **AT+CLEARSMS**

(this just clears the incoming SMS ready for next time)