

Infoalfa GPRS modem server

Product documentation v1.0

Release date: 21/3/2007

Table of contents

- Introduction 2
 - Message exchange scenarios 3
 - Service modules 3
- Installation 4
- Configuration and monitoring..... 6
 - Modem overview..... 7
 - Modem viewing 9
- Testing 12
- Sending messages to GPRS modem 14
- Alarm configuration 16
- Troubleshooting 18
 - Messages in Event viewer 18
 - Service problems 20
 - Network problems..... 21
- Licensing..... 22

Introduction

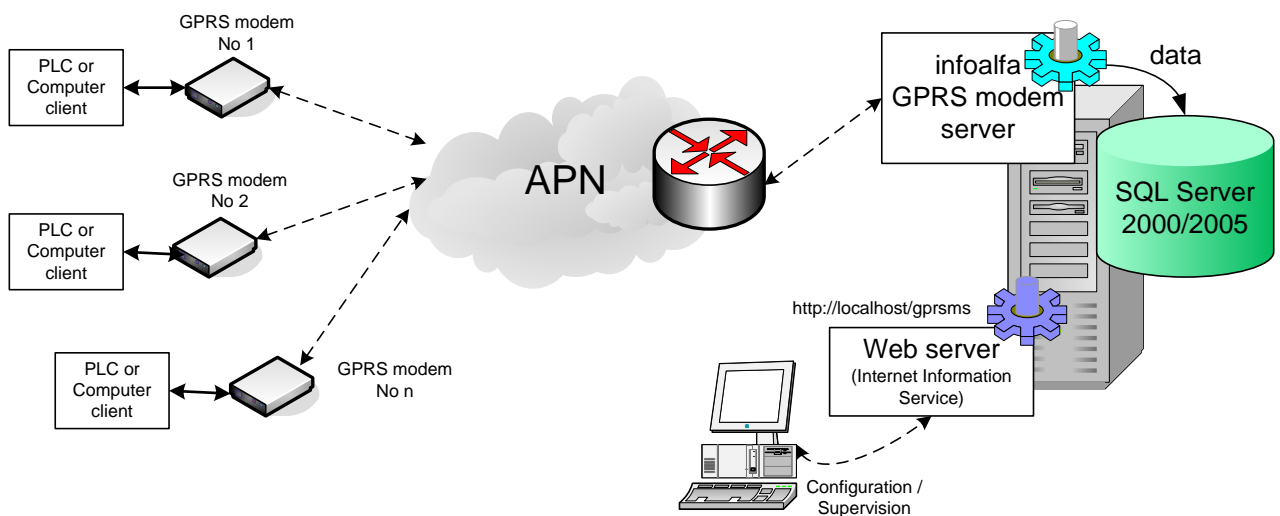
The purpose of this document is to explain what is Infoalfa GPRS modem server, how to install it and how to use it. For any additional question you may have, you may send your question on mail: gprsms@infoalfa.hr

The most interesting questions (and of course, answers) will be published on web page in „Q & A“ section. Check the web page <http://www.infoalfa.hr/gprsms>

Infoalfa GPRS modem server is software product developed in Microsoft .NET 2.0 Visual Basic, and it is released as a Microsoft Windows Service. It also uses Microsoft SQL database (2000 or 2005 version). Configuration is possible through .NET 2.0 aspx application or directly in database tables.

Event viewer is the best place for tracking messages that gprsms exchange with GPRS modems (which have microcontroller or even computer on the other side).

Messages that can be exchanged between server and clients (GPRS modems) are defined on SQL Server. However, some part of message can not be changed in this version of software (like 16-bit XOR checksum at the end of the message).



Message exchange scenarios

- A. client sends message to the server over modem
server responds with ACK (OK)
- B. client sends message to the server over modem
server receives incorrect or partial message and responds with ACK (NOK) or does not respond at all
client sends message again until receives ACK (OK)
- C. server sends message to client over modem
modem responds with ACK (OK), server updates status for the message from the queue
- D. server sends first message from the queue to client over modem
client does not respond until new timer event occur (set by *timerms* time in the registry) or responds with ACK (NOK)
server sends message from the queue again, until it receives ACK (OK) and then it updates status for the message from the queue
- E. server sends keepalive message, every (*KeepAlive * timerms*) milliseconds
Client receives message and does not respond.
If it responds with ACK (OK), this ACK could be misinterpreted (if previously server sent message from the queue and expects answer).

Service modules

Overview of main procedure in service is shown below:

Procedure	Description
OnStart	handles start of the service (initiated by the system)
OnStop	handles stop of the service (initiated by the system)
AcceptCallback	First async procedure, opens port 11000 and listens. After opening connection, creates <i>StateObject</i> which defines connection, initializes timer for asynchronous events and calls <i>AcceptCallback</i> again, opening port 11000 for new connection.
ReadCallback	This procedure is asynchronously called when message arrives, after connection is opened. It can launch <i>ParseMessage</i> procedure if arrived message is complete.
ParseMessage	Called by <i>ReadCallback</i> if it concludes that message is to be parsed.
StoreMessage	If whole message is parsed OK, then it calls this procedure, which constructs SQL command and inserts it into the database.
StateObject.Send	Synchronous procedure that begins sending message to modem.
StateObject.SendCallback	Asynchronous procedure which ends sending message to modem.
StateObject.CheckStatus	Asynchronous command called by the timer (<i>timerms</i> parameter in the registry). First it checks if connection is still alive. If not, shuts down the connection. It sends <i>KeepAlive</i> message if configured, then sends message from <i>SendMessageQueue</i> , if any.

Installation

Since there are few components of GPRS modem server, it is necessary to install:

a) Prerequisites (software that has to be installed before GPRS modem service)

Windows XP, 2000 or 2003 with .NET 2.0

SQL Server 2000 (or MSDE) or 2005 (or Express Edition)

b) GPRS modem server

GPRS modem server service (msi package installation)

- run “setup.exe”
- click “Next”
- Select destination folder for application. Default is “C:\Program Files\Infoalfa\GPRS modem server\”
- Confirm installation, click “Next”
- When username/password/confirm password popup screen appears, enter username, password and confirmation password for user that is used for iagrpsms service running. Note that, if using local account, type “.” before username (i.e. “.Administrator”)
- Wait while installation completes. If username/password was wrong, or user does not have administrator privileges, installation would not be successful. Try again starting “setup.exe”.
- “Installation Complete” screen appears if everything was OK. Click “Close”.

Note: during the installation of the service, all other needed files were copied in the folder structure:

Data	IAGRPSMS database attached to SQL Server
Service	iagrpsms modem server service
web	http://localhost/gprsms management application
utilities	test message exchange utility

install SQL Server IAGRPSMS database (with SQL Server script)

- You may run this script in the existing database or you may create new database (default IAGRPSMS).
- Open *iagrpsmsCreate.sql* script in the SQL Query Analyzer (SQL Server 2000) or SQL Server Management Studio (SQL Server 2005). This script can be found under utilities folder, under installation point (default is “C:\Program Files\Infoalfa\GPRS modem server\utilities”)
- Modify “create database” command on the begging of the script to create mdf/ldf files for the database on desired location. Default is “C:\Program Files\Infoalfa\GPRS modem server\Data”.
- If you want to create database objects in the existing database, you have to delete “create database” and “alter database” commands from the script.
- Run the script.
- Watch the results window if there are any error (messages in red color).

install gprsms web configuration application:

- Open *Control Panel* → *Administrative Tools* → *Internet Information Services*

- Expand (*local computer*), expand *Web Sites*.
- Right click on *Default web site*, and then click *New, Virtual Directory*. “*Directory Creation Wizard*” appears.
- Click “Next”, type “gprsms” for Alias, then click “Next”.
- Enter subfolder “web” under installation point you choose during the setup (default is “C:\Program Files\Infoalfa\GPRS modem server\web”). Click “Next”.
- Click “Next” again. “You have successfully completed the Virtual Directory Creation Wizard” screen appears. Click “Finish”.
- Right click on “gprsms” under “Default Web Site”, and then click “Properties”.
- Select “Documents” tab. Remove all default document listed (usually *Default.htm, Default.asp, index.htm, iisstart.asp*). Click “Add”, enter “PeanutLogon.aspx”. Click “OK”, then “OK” again.
- If necessary, edit *web.config* file (under web subfolder) to configure server and the database that web application is connected to:

```

...
<appSettings>
  <add key="Version" value=""/>
  <add key="SqlCon1.Server" value="localhost"/>
  <add key="SqlCon1.Database" value="IAGPRSMS"/>
  <add key="CrystalImageCleaner-AutoStart" value="true"/>
  <add key="CrystalImageCleaner-Sleep" value="60000"/>
  <add key="CrystalImageCleaner-Age" value="120000"/>
</appSettings>
...

```

c) Test utilities

gprsmodem.exe does not need any installation. See “Testing” section to learn how to use this utility.

For testing purpose, you may also use *gprsmodem.exe* utility.

Note that web application is generated with *Infoalfa Peanut Form Generator* and that license for runtime module of this generator has restrictions on application name (i.e. *gprsms*).

For more details you may send your question on e-mail gprsms@infoalfa.hr

Configuration and monitoring

To exchange any message between server and client, you have to configure server service through web based application (<http://localhost/gprsms>) or directly in SQL Server tables (not recommended).

Messages can be formatted as you need. However, there are some restrictions in formats.

Message format

Byte	Content
0	Constant 0x02 (start of text)
1	Constant 0x00
2	Address lower byte
3	Address higher byte
4	Block number (0x00 - 0xff)
5	Message length (n)
...	
n-1	XOR, lower byte
n	XOR, higher byte

ACK message

Byte	Content
0	Constant 0x02 (start of text)
1	Address lower byte
2	Address higher byte
3	0xff = message transferred OK, 0x00 = message not transferred or incorrect transfer (NOK)
4	XOR, lower byte
5	XOR, higher byte

Log on to admin application on <http://localhost/gprsms>

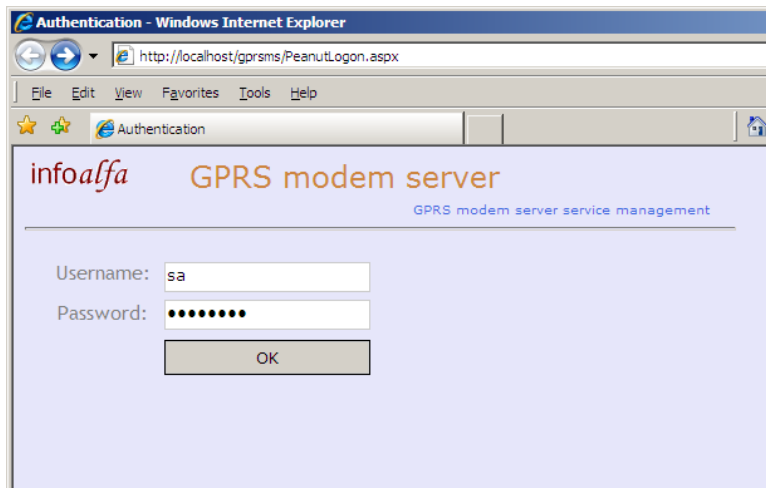
If you installed web application on some other location, simply replace “localhost” with name or IP address of server where application is installed.

You can install web application on any server on network, but you must reconfigure “web.config” file in folder “web”:

```
...
    <appSettings>
        <add key="Version" value=""/>
        <add key="SqlCon1.Server" value="localhost"/>
        <add key="SqlCon1.Database" value="IAGPRSMS"/>
    ...
```

Replace “localhost” with name of server where SQL Server is installed, and the name of the database “IAGPRSMS” replace with actual name.

With sa or any other privileged user name on SQL Server, you can log on to application:

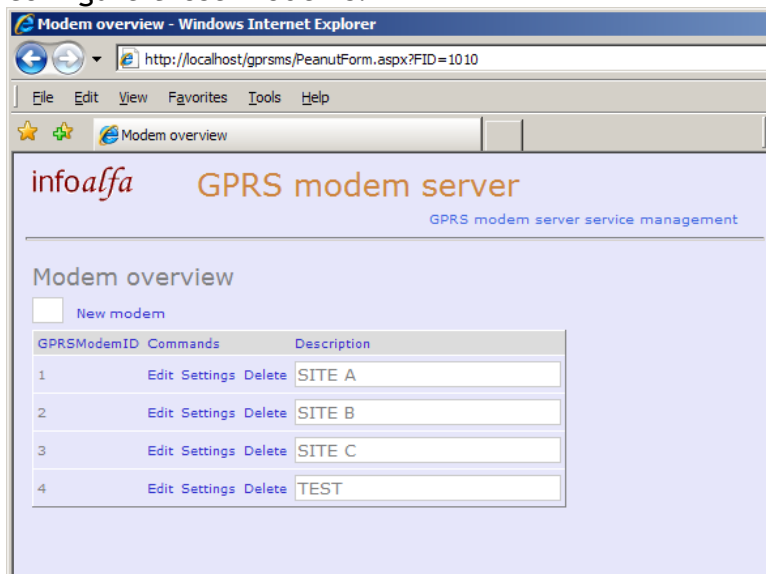


After successful logon, you can see main menu:



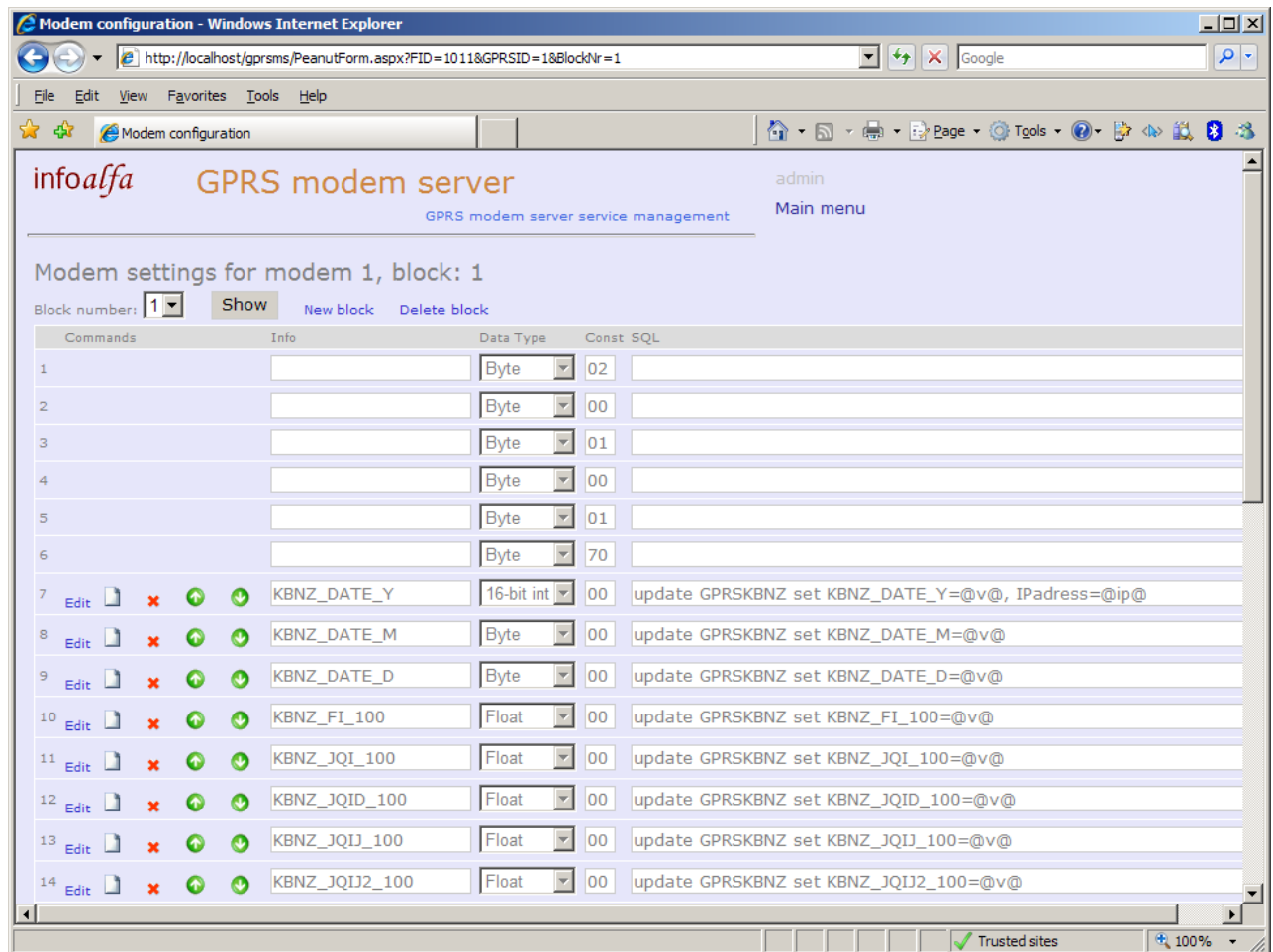
Modem overview

In modem overview, you may create new modem, see the list of all configured modems or configure those modems:



With “Edit” you may edit name of the modem. With “Delete” you may delete modem settings. Confirmation of deletion is needed.

“Settings” opens form with these settings for modem:







On the top left part of the window you may select block number and then click button “Show”. You may create new block with linkbutton “New block” or delete block with linkbutton “Delete block”. Confirmation of deletion is needed.

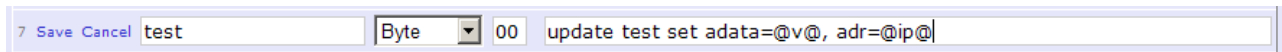
Table with message definition for modem is shown at the center of the screen. First 6 lines are created automatically when you hit “New block” and can’t be changed through application. 6th byte (length of message) is updated automatically, as you enter definition of each data, line by line.

Last data in the message is 16-bit XOR and that also can not be changed.

On the left side of each message you may find commands:

Edit	Change selected line in edit mode
New ()	Inserts new line after selected
Delete ()	Deletes selected line (warning: no confirmation is needed)
Up ()	Moves selected line up
Down ()	Moves selected line down

In edit mode, you may change Info column, Data type column, Const column or SQL column.



On the left side appears “Save” command, which confirms change and “Cancel” command, which discards commands.

At the dropdown menu in the “Data Type” column you may select Type of Data:

- Byte (8-bit)
- Float (32 bits)
- 32-bit int
- 16-bit int

“Const” column is not parsed, it is used only in header of the message (first 6 bytes). SQL column can be used to type SQL query, which can be used to update any table in the database. Before executing, @v@ is replaced with actual value in the message and @ip@ is replaced with IP address of the modem.

Parser parses line by line, so it is possible that one message generates many SQL queries.

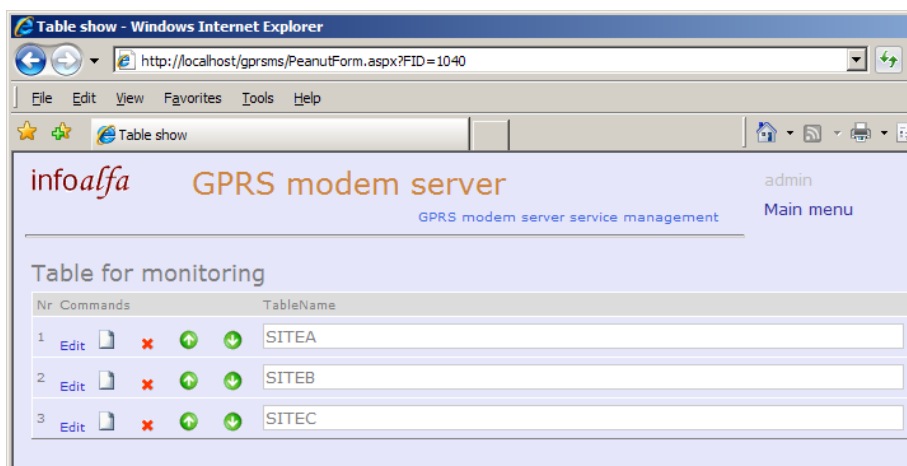
When the message is parsed, it is expected that lowest byte is stored first.

Modem viewing

In this section of main menu, you may find configuration form and form for monitoring.

Configuration form enlists tables which are to be shown on the monitoring form. Each line has commands for edit, new, delete (warning: no confirmation), moving up and down, as well as SQL Table name in the database.

Tables and modems are not necessarily related 1:1, so it is possible to have more tables than modems in the monitor window and vice versa.



Modem monitoring is shown in the shape of multiline tables, which has enhanced name and semaphore (green or red) which indicates if any message has been received in the last 15 minutes. If any column in the table has been touched, trigger on the table updates “LastMessage” field with current date and time.

The screenshot shows a web browser window titled "Modem monitoring - Windows Internet Explorer". The address bar shows the URL "http://localhost/gprsms/PeanutForm.aspx?FID=1050". The page content includes the "infoalfa GPRS modem server" logo and navigation links for "admin" and "Main menu". The main section is titled "Modem monitoring" and contains three tables:

SITEA	LastMessage	Alarm	KBNZ_DATE_Y	KBNZ_DATE_M	KBNZ_DATE_D	KBNZ_FI_100	KBNZ_JQI_100	KBNZ...
●	13-03-2007 11:22:41	1	2006	11	27	9.39712	7.044	0.79
KBNZ_JQI2_100	KBNZ_JQI3_100	KBNZ_JQI4_100	KBNZ_JQI5_100	KBNZ_JQI6_100	KBNZ_JQI7_100	KBNZ_JQI8_100	KBNZ_JQI9_100	KBNZ...
2.14991	1.95892	0	0	0	0	0	0	0
KBNZ_CFQID_100	KBNZ_CFQI1_100	KBNZ_CFQI2_100	KBNZ_CFQI3_100	KBNZ_CFQI4_100	KBNZ_CFQI5_100	KBNZ_CFQI6_100	KBNZ_CFQI7_100	KBNZ...
63.8645	187.7	188.099	161.32	0	0	0	0	0
KBNZ_CFQI10_100								
0								
SITEB	LastMessage	LUCKO_DATE	LUCKO_FI_100	LUCKO_JQI_100	LUCKO_JQID_100	LUCKO_JQI1_100	LUCKO_JQI2_100	LUCKO...
●		0						
LUCKO_JQI5_100	LUCKO_JQI6_100	LUCKO_JQI7_100	LUCKO_JQI8_100	LUCKO_JQI9_100	LUCKO_JQI10_100	LUCKO_CFQI_100	LUCKO_CFQID_100	LUCKO...
LUCKO_CFQI3_100	LUCKO_CFQI4_100	LUCKO_CFQI5_100	LUCKO_CFQI6_100	LUCKO_CFQI7_100	LUCKO_CFQI8_100	LUCKO_CFQI9_100	LUCKO_CFQI10_100	
SITEC	LastMessage	SRCM_DATE	SRCM_FI_100	SRCM_JQI_100	SRCM_JQID_100	SRCM_JQI1_100	SRCM_JQI2_100	SRCM...
●		0						
SRCM_JQI5_100	SRCM_JQI6_100	SRCM_JQI7_100	SRCM_JQI8_100	SRCM_JQI9_100	SRCM_JQI10_100	SRCM_CFQI_100	SRCM_CFQID_100	SRCM...
SRCM_CFQI3_100	SRCM_CFQI4_100	SRCM_CFQI5_100	SRCM_CFQI6_100	SRCM_CFQI7_100	SRCM_CFQI8_100	SRCM_CFQI9_100	SRCM_CFQI10_100	

Important note: alarms have to be configured as described in the chapter “Alarm configuration”

Registry entries HKEY_LOCAL_MACHINE\SOFTWARE\Infoalfa

Some variables which can be changed with **regedit** utility are listed below:

key	Explanation	Default
port	TCP port for message exchange	11000
Database	database with service system and control tables configured via web application	IAGPRSMS
SQLServer	name or IP address of computer with SQL Server database	(local)
timerms	timer counter in ms (1000ms=1sec)	5000
KeepAlive	counter of keepalive messages. If 0, keepalive message would not be sent. Keepalive message is defined in table [KeepAliveMessage], field [Message]	0

Note that you have to restart service (through Services utility) to take new values in effect.

Testing

To test Infoalfa GPRS modem server configuration, you may use gprsmodem emulator. This is command line utility, which connects to server service and exchange messages. It can be used to test basic communication, but not advanced messages.

It is supposed to be copied to “utilities” folder under “Infoalfa”, but it also can be copied anywhere else on hard disk.

command syntax is:

```
gprsmodem <address>, <input file name>, [<ACK file name>]
```

where <address> is IP address or name of computer where Infoalfa GPRS modem service is running (if on local computer, you may type 127.0.0.1 or localhost),

<input file name> is file name of input data that modem has to send to server.

<ACK file name> is optional; it can be used if you want GPRS modem emulator answers to server with ACK message.

Input file with message has to be in text format, as a list of decimal or hexadecimal data separated with semi-colons.

In this example you may see test message for modem on address 0x0001, block 1, with total length of 100 bytes (98 bytes included in the file):

inputmessage.txt

```
2; &h0; &h1; &h0; &h1; &h64; &hC7; &h71; &h9C; &h3F; &h55; &h55; &h15; &h40; &hC7; &h71; &h5C; &h40; &h1C; &hC7; &h91; &h40; &h55; &h55; &hB5; &h40; &h8E; &hE3; &hD8; &h40; &hC7; &h71; &hFC; &h40; &hFF; &hFF; &hF; &h41; &hBD; &h9D; &h11; &h41; &h1C; &hC7; &h21; &h41; &h39; &h8E; &h33; &h41; &h97; &hB7; &h43; &h41; &h0; &h0; &h80; &h3F; &h55; &h99; &h9B; &h3F; &h5C; &h2B; &h0; &h3E; &hCA; &hB8; &h52; &h35; &h89; &hE4; &h8A; &hD0; &h24; &h41; &h20; &hBE; &h64; &h35; &hD7; &h62; &hE0; &h1B; &h0; &h0; &h4; &h39; &h32; &h56; &hA5; &hF0; &h29; &h4C; &h73; &h6A; &h4A; &h57
```

gprsmodem automatically adds correct 16-bit XOR at the end of message (it should not be included in input file).

If you try to connect to server where service is not running on port 11000, you will get error message:

```
C:\iagprsms\utilities>gprsmodem localhost i.txt
No connection could be made because the target machine actively refused it
```

If you have iagprsms service running on machine you have asked for, you will get few lines of conversation between modem and server:

```
C:\iagprsms\utilities>gprsmodem localhost i.txt
Output message 100 bytes: 2 0 1 0 1 64 C7 71 9C 3F 55 55 15 40 C7 71 5C 40 1C C7
91 40 55 55 B5 40 8E E3 D8 40 C7 71 FC 40 FF FF F 41 BD 9D 11 41 1C C7 21 41 39
8E 33 41 97 B7 43 41 0 0 80 3F 55 99 9B 3F 5C 2B 0 3E CA B8 52 35 89 E4 8A D0 2
4 41 20 BE 64 35 D7 62 E0 1B 0 0 4 39 32 56 A5 F0 29 4C 73 6A 4A 57 60 BB
Sending data.
Received 6 bytes: 2 1 0 FF 2 FE
^C
```

On this list you can see your message from file listed above that is completed with 16-bit XOR at the end of message and sent to server, which responded with ACK message.

Program does not stops, it waits for new messages (if any) indefinitely, so you may hit CTRL-C to stop program.

If you have configured KeepAlive option, you may wait to see keepalive message configured in the database:

```
Output message 100 bytes: 2 0 1 0 1 64 C7 71 9C 3F 55 55 15 40 C7 71 5C 40 1C C7
91 40 55 55 B5 40 8E E3 D8 40 C7 71 FC 40 FF FF F 41 BD 9D 11 41 1C C7 21 41 39
8E 33 41 97 B7 43 41 0 0 80 3F 55 99 9B 3F 5C 2B 0 3E CA B8 52 35 89 E4 8A D0 2
4 41 20 BE 64 35 D7 62 E0 1B 0 0 4 39 32 56 A5 F0 29 4C 73 6A 4A 57 60 BB
Sending data.
Received 6 bytes: 2 1 0 FF 2 FE
Received 4 bytes: 48 41 4C 4F
^C
```

In this example, we configured KeepAlive message in the database to response with "HALO" (hexadecimal 0x48, 0x41, 0x4C, 0x4F).

And in the next example you may see modem emulator to send message to server, then server responded with ACK, and then modem responses with ACK (which is not how it is meant to be, but server does not mind for this last ACK). Finally, we can see few KeepAlive messages responded by modem with ACK:

```
C:\iagprsms\utilities>gprsmodem localhost i.txt ack.txt
Output message 114 bytes: 2 0 1 0 1 70 D6 7 B 1B 9F 5A 16 41 73 68 E1 40 75 1 4B
3F 99 4 9 40 B 98 9 40 DA BD FA 3F 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 CB 4C 16 44 44 75 7F 42 20 B3 3B 43 49 19 3C 43 E0 51 21 43 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Sending data.
Received 6 bytes: 2 1 0 FF 2 FE
Output message 6 bytes: 2 1 0 FF 2 FE
Sending data.
Received 4 bytes: 48 41 4C 4F
Output message 6 bytes: 2 1 0 FF 2 FE
Sending data.
Received 4 bytes: 48 41 4C 4F
Output message 6 bytes: 2 1 0 FF 2 FE
Sending data.
^C
```

Sending messages to GPRS modem

You may send message to GPRS modem that is connected by inserting appropriate command to table `SendCommandQueue`.

Column name	Type	Description
id (PK)	int	primary key, identification of message
GPRSModemID	varchar(20)	GPRS modem ID (as defined in configuration) or current IP address
State	tinyint	status of message: 1=inital status (ready to send) 2=message sent, waiting for ACK 3=messege sent, ACK received
InsertDateTime		Date and time of data insertion
LastStateDateTime		Last status date and time
Message	varbinary(4000)	Body of message in binary format. Note that

Service fetches data from table `SendCommandQueue` using view `vSendCommandQueue`. When it sends message for the first time for each modem, it changes intital status (expected 1) to 2. Next time service fetches data again from this table and sends again all messages for each modem with status 1 or 2. That means that you have to configure parameter `timerms` to let PLC (or computer) on the other side enough time to send ACK message. When service receives ACK, it changes status for the last message in status 2 to 3 and when next time timer event occurs, it sends message with minimum id in status 1 for each modem.

Since `GPRSModemID` is typically modem number configured through <http://localhost/gprsms> application, you may use this number to identify modem, but sometimes it is usefull to use IP address if you want to send message to specific connected modem, regardless of it's number.

Example of sending message to modem number 64:

```
insert into SendCommandQueue
select (select max(id)+1 from SendCommandQueue), 64, 1, getdate(), getdate(),
convert(varbinary(4000), char(2)+char(0)+char(64)+char(0)+char(1)+char(10)+'OK')
```

16-bit XOR checksum is automatically added at the end of the message, so you don't have to put it in the message. Header is not constructed by the service, so you have to build it with first 6 bytes (as shown above). When you calculate message length, you should add 2 bytes for 16-bit XOR checksum.

To test messages sent to modem by GPRS modem server, you may use `gprsmodem` utility described in the chapter "Testing".

Sometimes it is useful to have separate tables for each modem, from which you can build a message which has to be sent to GPRS modem. In that case you can do the following:

create appropriate table, which consists of all columns with data to send to GPRS modem

```
CREATE TABLE [GPRSOUTMsg] (  
    [f1] [int] NULL ,  
    [f2] [float] NULL  
) ON [PRIMARY]  
GO
```

add update trigger to the table, which inserts all the data from the table into the table SendCommandQueue

```
CREATE TRIGGER trgMsgInsert ON [dbo].[GPRSOUTMsg]  
FOR UPDATE  
AS  
exec spSetOutMessage 'GPRSOUTMsg', 64, 16
```

and the job is done!

Trigger uses stored procedure

```
spSetOutMessage <table name>, <modem number>, <block number>
```

which composes message and inserts it into the SendMessageQueue.

Now, when you want to send a message to the GPRS modem, simply update GPRSOUTMsg table with data you want to send to modem.

```
insert into GPRSOUTMsg(f1, f2) values(10, 1.234)
```

and message will be sent when next timer event occurs on GPRS modem server.

Content of the table SendMessageQueue before sending message is:

id	GPRSModemID	State	InsertDateTime	LastStateDateTime	Message
14	64	1	2007-03-14 12:53:06.717	NULL	0x0200400001140A0000005839B4C876BEF33F

Alarm configuration

If you want to set alarm indicator to be displayed as shown on „Modem monitoring“ form in application <http://localhost/gprsms>, you have to create job that runs *spSetAlarm* procedure that is scripted and installed during the installation of IAGPRSMS database. Installation script for that job:

```
USE [msdb]
GO
/***** Object: Job [Set Alarm] Script Date: 03/15/2007 12:13:29 *****/
BEGIN TRANSACTION
DECLARE @ReturnCode INT
SELECT @ReturnCode = 0
/***** Object: JobCategory [[Uncategorized (Local)]] Script Date: 03/15/2007 12:13:29 *****/
IF NOT EXISTS (SELECT name FROM msdb.dbo.syscategories WHERE name=N'[Uncategorized (Local)]' AND
category_class=1)
BEGIN
EXEC @ReturnCode = msdb.dbo.sp_add_category @class=N'JOB', @type=N'LOCAL', @name=N'[Uncategorized
(Local)]'
IF (@@ERROR <> 0 OR @ReturnCode <> 0) GOTO QuitWithRollback

END

DECLARE @jobId BINARY(16)
EXEC @ReturnCode = msdb.dbo.sp_add_job @job_name=N'Set Alarm',
        @enabled=1,
        @notify_level_eventlog=0,
        @notify_level_email=0,
        @notify_level_netsend=0,
        @notify_level_page=0,
        @delete_level=0,
        @description=N'No description available.',
        @category_name=N'[Uncategorized (Local)]',
        @owner_login_name=N'admin', @job_id = @jobId OUTPUT
IF (@@ERROR <> 0 OR @ReturnCode <> 0) GOTO QuitWithRollback
/***** Object: Step [spSetAlarm] Script Date: 03/15/2007 12:13:29 *****/
EXEC @ReturnCode = msdb.dbo.sp_add_jobstep @job_id=@jobId, @step_name=N'spSetAlarm',
        @step_id=1,
        @cmdexec_success_code=0,
        @on_success_action=1,
        @on_success_step_id=0,
        @on_fail_action=2,
        @on_fail_step_id=0,
        @retry_attempts=0,
        @retry_interval=0,
        @os_run_priority=0, @subsystem=N'TSQL',
        @command=N'spSetAlarm',
        @database_name=N'IAGPRSMS',
        @flags=0
IF (@@ERROR <> 0 OR @ReturnCode <> 0) GOTO QuitWithRollback
EXEC @ReturnCode = msdb.dbo.sp_update_job @job_id = @jobId, @start_step_id = 1
IF (@@ERROR <> 0 OR @ReturnCode <> 0) GOTO QuitWithRollback
EXEC @ReturnCode = msdb.dbo.sp_add_jobschedule @job_id=@jobId, @name=N'Sch1',
        @enabled=1,
        @freq_type=4,
        @freq_interval=1,
        @freq_subday_type=4,
        @freq_subday_interval=3,
        @freq_relative_interval=0,
        @freq_recurrence_factor=0,
        @active_start_date=20061210,
        @active_end_date=99991231,
        @active_start_time=0,
        @active_end_time=235959
IF (@@ERROR <> 0 OR @ReturnCode <> 0) GOTO QuitWithRollback
EXEC @ReturnCode = msdb.dbo.sp_add_jobserver @job_id = @jobId, @server_name = N'(local)'
IF (@@ERROR <> 0 OR @ReturnCode <> 0) GOTO QuitWithRollback
COMMIT TRANSACTION
GOTO EndSave
QuitWithRollback:
        IF (@@TRANCOUNT > 0) ROLLBACK TRANSACTION
EndSave:
```

You have to change username „admin“ with appropriate username (marked with red color) and name of the database (if not default, i.e. „IAGPRSMS“, also marked with red color).

Procedure *spSetAlarm* „walks“ through all tables defined in table TableShow (managed through form „Configuration“ under „Modem monitoring“ and updates Alarm indicator to 1 if LastMessage is greater than 15 minutes.

You can change this period by modifying *val* field in the table *Config*, where *param*='AlarmTimeOut'.

On every table that you have to monitor „Alarm“ indicator, you have to add appropriate trigger to the table (in this example table name is „TEST“):

```
ALTER TRIGGER [dbo].[trgTESTLastMessage]
ON [dbo].[TEST]
AFTER INSERT,UPDATE
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    declare @AlarmBefore tinyint, @AlarmAfter tinyint

    select @AlarmBefore=min(Alarm) from deleted
    select @AlarmAfter=min(Alarm) from inserted

    if @AlarmBefore=@AlarmAfter
    update TEST set LastMessage=getdate()
END
```

Trigger assumes that only procedure *spSetAlarm* changes Alarm indicator, so it skips updating *LastMessage* field in that case. Otherwise procedure *spSetAlarm* would change *Alarm* indicator every time it changes *Alarm*.

Troubleshooting

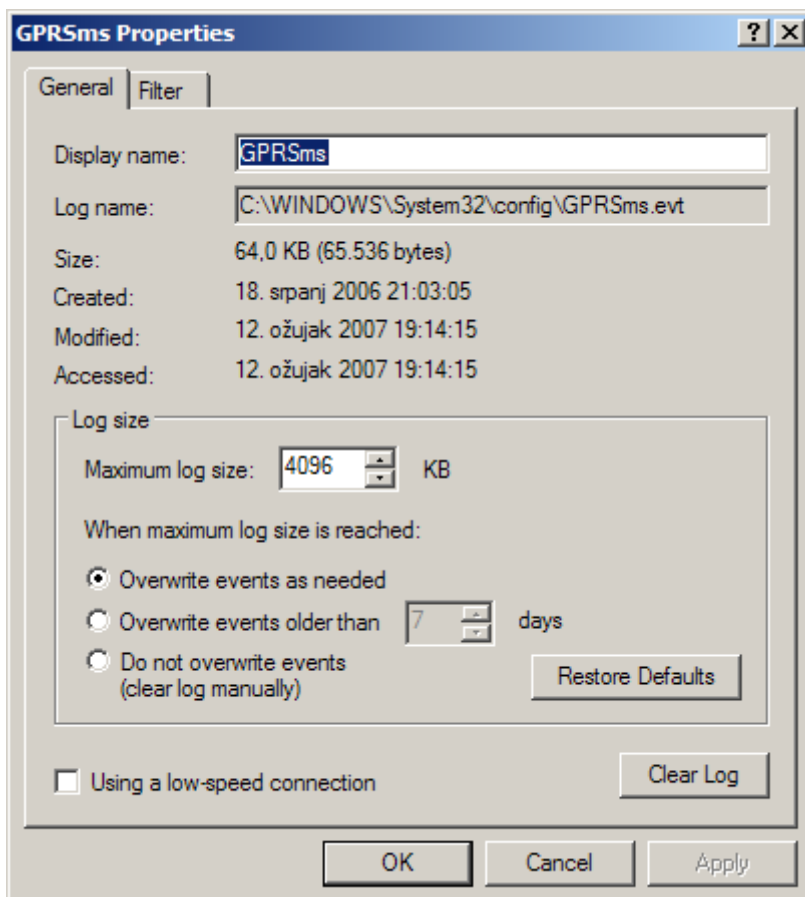
Messages in Event viewer

To view *Infoalfa GPRS modem service* log, you can open event viewer through Computer Management or
Control Panel → Administrative Tools → Event Viewer.

In the the „System“ section you may find information about starting and stoping of the service (like any other Windows service), as well as catastrophic failures of *iagprsms* service.

All other messages are stored in the special section named „GPRSms“. Source column tells you about module of the service which registered message (overview of the modules are given in the chapter „Introduction“ of this document), and the body of the message can be read when you click on the line containing message.

Since service writes a lot of messages in the log, it is recommended that you expand size of the event viewer, and set „Overwrite as needed“ option.



Messages are usefull to track problems with connections and messages.

In the table below, you may find tracking messages that are written in the event viewer, grouped by modules.

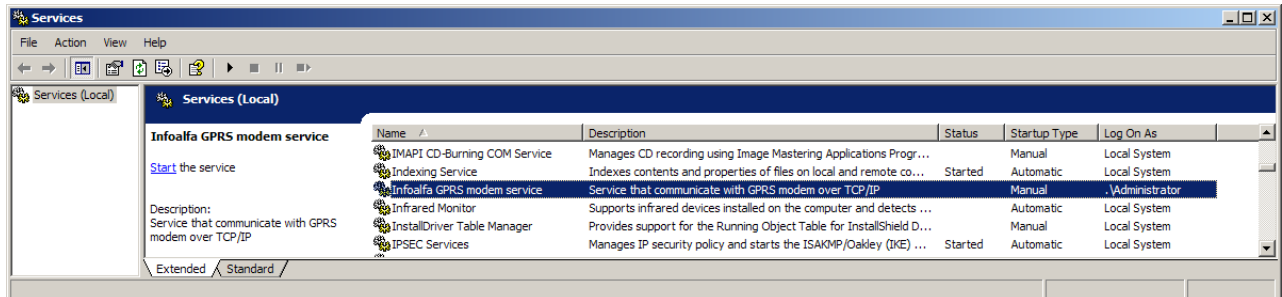
Procedure	Error list
OnStart	Parameters - port: [port number], Database: [database name], SQLServer:[IP address or network name of the server], timerms: [time in ms], debug=[true or false], KeepAlive=[n]
	There are [n] modems: [numbered list]
	Waiting for a first connection...
	"Error in starting service." [error description]
OnStop	Service iagprsms stoped sucessfully
AcceptCallback	Established connection with [ip address of the modem]
	Waiting for a connection...
	[error description]
ReadCallback	First packet from client (modem [n]) Read [n] bytes from socket Data : [xx xx xx xx xx xx]
	[error description]
	[IP address] (modem [n]) ACK message received.
	[IP address] (modem: [n]) ACK message, *** BAD XOR *** expected: [xxxx]
ParseMessage	[IP address] (modem [n]) Message length [n] must be >=8"
	[IP address] (modem [n]) Partial Data : [dd dd dd dd dd dd]
	[IP address] (modem [n]) [xx xx xx xx] Data : [dd dd dd dd dd dd] Calculated XOR: [xxxx] Received XOR at offset [n] : [xxxx]
	[IP address] (modem [n]) Error in storing message
	[IP address] (modem [n]) Calculated output XOR: [xx]
	[error description]
	Wrong constant on position [n]: [dd], expected: [xx]
	[IP address] (modem [n]) Output data: [xx xx xx xx]
	[IP address] (modem [n]) Sent [n] bytes to client.
	[IP address] (modem [n]) Error message: [error description]
StoreMessage	Wrong constant on position [n]: [dd], expected: [xx]
StateObject.Send	[IP address] (modem [n]) Output data: [xx xx xx xx]
StateObject.SendCallback	[IP address] (modem [n]) Sent [n] bytes to client.
StateObject.CheckStatus	Sending keep alive message
	Closed connection with [IP address] (modem: [n])
	Sending numbered command for modem: [n]
	[IP address] (modem [n]) Error message: [error description]

Service problems

If you have trouble with Infoalfa GPRS modem service, you should do following checking's:

Check if Infoalfa GPRS modem service is running, in the

Control Panel → Administrative Tools → Services



You may start the service with right click on the line shown on picture above, then click Start. You may also configure service to run automatically on startup (right click, Properties, Startup Type).

Check error messages in the event viewer, as described in the previous section.

When the service is started, you may see listener with netstat -a command in the command window (among other lines):

```
TCP        saturn:11000                saturn.infoalfa.local:0    LISTENING
```

where saturn is the name of the server.

If not, check if you SQL Server is running. In that case, start database first and then restart service.

Check event viewer. If you find message like this:

```
Error in starting service:An error has occurred while establishing a connection to the server. When connecting to SQL Server 2005, this failure may be caused by the fact that under the default settings SQL Server does not allow remote connections. (provider: Named Pipes Provider, error: 40 - Could not open a connection to SQL Server)
```

You can be sure that your database is not up and running.

Network problems

After establishing several connections, you may see those connections in the command window with netstat -n command:

```
C:\Documents and Settings\Administrator>netstat -n
```

```
Active Connections
```

Proto	Local Address	Foreign Address	State
TCP	10.220.200.45:1433	10.220.200.41:2609	ESTABLISHED
TCP	10.220.200.45:1977	10.220.200.40:139	ESTABLISHED
TCP	10.220.200.45:3389	10.220.200.65:1281	ESTABLISHED
TCP	10.220.200.45:11000	10.220.230.196:63549	ESTABLISHED
TCP	10.220.200.45:11000	10.220.230.197:12438	ESTABLISHED
TCP	10.220.200.45:11000	10.220.230.198:9399	ESTABLISHED
TCP	10.220.200.45:11000	10.220.230.199:14147	ESTABLISHED
TCP	10.220.200.45:11000	10.220.230.200:7421	ESTABLISHED
TCP	10.220.200.45:11000	10.220.230.201:63550	ESTABLISHED
TCP	10.220.200.45:11000	10.220.231.31:5845	ESTABLISHED
TCP	10.220.200.45:11000	10.220.231.175:25520	ESTABLISHED
TCP	10.220.200.45:11000	10.220.231.176:18066	ESTABLISHED

All connections on port 11000 are active GPRS modem service connections (10.220.200.45 is the IP address of the server).

If you have problem that some connections stay in “established” state, and the connection is broken, configure KeepAlive message with the value greater than zero and define keep alive message in the database table KeepAliveMessage. Every time server tries to send keepalive message, it will shutdown dead connection (in the state CLOSE_WAIT or even ESTABLISHED).

The same effect you will get if you manually try to send message to the modem over dead connection.

Licensing

After installing trial version of Infoalfa GPRS modem server, you may use this product for 30 days. After that period, it would not be possible to start service.

In the event viewer you will find message

“Evaluation period is finished. Please find more info at <http://www.infoalfa.hr/gprsms>”

You have to order your license for this product. More information you may find on web page <http://www.infoalfa.hr/gprsms> or you may ask any question about this product on mail gprsms@infoalfa.hr