CLOGGER Hardware Installation and Configuration Guide V3.0

Table of Contents

Introduction

The *CLOGGER* enables the access control and remote monitoring of photocopiers. It can be connected directly to the photocopier and thus controlling entirely the photocopier, see Figure 1, or it can be used as a transparent interface (spy) between the photocopier and a third party access control system for remote monitoring only, *see* Figure 2. The *CLOGGER* integrates a GSM/GPRS modem and makes available usage statistics, alerts, and errors to the monitoring center. The monitoring center has full remote control of the *CLOGGER*, i.e. enabling access control to the photocopier or disabling it, setting up parameters, etc…

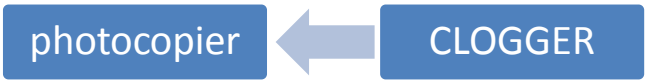


Figure 1

Figure 2

An RF module is integrated for wireless data transfer, thus insuring the remote connection to the monitoring center.

Using the USB RF module shown below we have full remote control of the CLOGGER.

Hardware overview

Features:

- MCU: LPC1763 Cortex M3, 32bits, 100 MHz, 256KB Flash, 64KB RAM
- GSM/GPRS module 900/1800/1900Mhz with SIM holder
- Fully integrated Sub-GHz RF transceiver supporting the 433/868/915MHz ISM frequency band
- RS232 port
- DB15 Female connector with built in male adapter
- Three isolated digital inputs (AC Input optocouplers)
- Two fully sealed and Ultra-miniature relays 1A/125 VAC
- External RTC with backup battery
- Temperature sensor
- Buzzer
- Three status leds
- Three configuration buttons
- Reed switch
- Reset circuit
- High efficient switching mode power supply (SMPS)
- FR-4, 1.5 mm, green solder mask
- Dimensions:113x86mm

Hardware configuration:

Reset Jumper (JP2):

Close this jumper and then open it to restart the system.

Boot loader Jumper (JP3):

Close this jumper and reset the system.
The system is forced to boot mode and ready for programming.
Use flash magic to upload a new firmware.

Shunt setup Jumper (JP12):

For normal operation, this jumper should be closed.

Modem power Jumper (JP4):

This jumper should be closed. It powers up the modem.

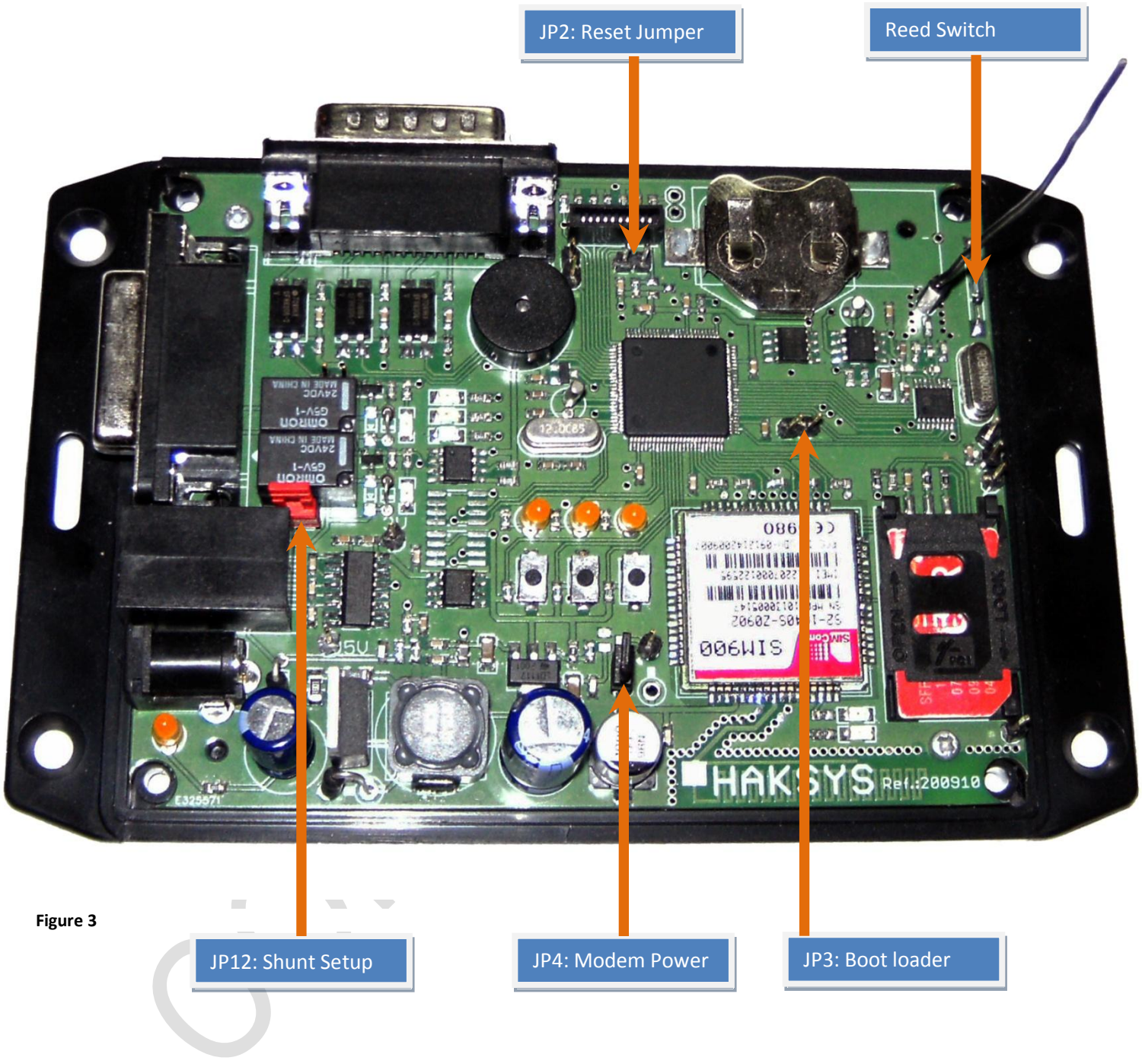

JP2: Reset Jumper
Reed Switch
JP12: Shunt Setup
JP4: Modem Power
JP3: Boot loader

Figure 3

Installing and connecting the device

The *CLOGGER* must be installed by a photocopier specialist. It is seen by the photocopier as a key counter and must be connected as so.
The *CLOGGER* can be interfaced to different models of photocopiers, like those from Xerox, Ricoh, Canon, Kyocera-Mita, Océ, Panasonic, Sharp and Toshiba. It can be adapted easily to new models.
For each model, a dedicated interfacing cable is needed.
Powering up the *CLOGGER* can be done in 3 ways:

- Directly from the +24V photocopier power supply.
- From the +24V power supply available on one of the counter pulse terminals of the photocopier.
- From an external power supply (+24V & 500 mA).

Accessing the device

The *CLOGGER* responds to a predefined list of commands. These commands are used for configuration & control, status or execution of specific tasks. For more information, see "Commands set Summary" section.
There are 3 ways to access the *CLOGGER*

- ***Console***:
  The Console port on the *CLOGGER* gives access to it through a serial cable connected to a workstation or terminal. By typing commands on a terminal or in a terminal-emulation program of a workstation, the *CLOGGER* gives the adequate responses. For more information, see "Using the Console Connection" section.
- ***Wireless*:**
  The console interface can be accessed remotely to execute the commands and get the responses from the *CLOGGER* which simplifies servicing the device. For more information, see "Using the wireless Connection" section.
- ***SMS*:**
  The monitoring center has full control of the *CLOGGER*. The Commands are encapsulated in preformatted SMS messages, executed by the *CLOGGER* and the response is sent back. For more information, see "Using the SMS connection" section.

Commands format

A command is a string of ASCII characters. It consists of a keyword, followed if necessary by parameters. The delimiter is the space character <*sp*>

Example:

Make 3 beeps of 10 milliseconds.

```
beep<sp>3<sp>10
```

Keyword: beep Parameter1: 3 Parameter2: 10

The *CLOGGER* parses the command, executes it, and sends a response if any.

Using the console connection[1]

Console settings:

- baud rate: 115200
- parity: None
- data bits: 8
- stop bit: 1
- flow control: None

Opening a console session:

After launching a serial terminal-emulation program (Hyperterminal for example) on a workstation and typing the Carriage return key (↵) "ENTER", a prompt "$" is displayed and the *CLOGGER* is ready to accept commands.

[1] *HAKSYS has conceived a special cable under the ref. CBL001 to establish a console session.*

Examples:

```
$version
HAKSYS CLOGGER V1.00
Nov 15 2010 20:47:15 X[7B982CAF]
$time
17:05:59
$date
22/11/2010
$datetime
22/11/2010 06:12:39 (47) (326)
$
```

Using the wireless connection

A USB module has been conceived. It integrates a Sub-GHz RF transceiver supporting the 433/868/915MHz ISM frequency band to be connected directly to the workstation. This module enables remote data access between the CLOGGER and the workstation.

The *CLOGGER* receives the RF messages, parses them, executes them, and sends the response if any.

All the commands from the console starting with "rf group" are directed to the module witch sends them, and prints the response back to the console.

Using the SMS connection

The *CLOGGER* has a built in GSM/GPRS module. It receives preformatted SMS messages, parses them, executes them, and delivers a report to the monitoring center.

SMS command and response messages formats:

The SMS command may include multiple commands separated by a semicolon ";".

Command:

Response:

Or,

Command:

Response:

When using the "exec" key word, the response message omits the header.

Examples:

Simple command message: beep

SMS command message:

```
execute 0123 beep;
```

SMS response message:

```
HAKSYS CLOGGER V1.00 28D49259 000002 22/11/2010 06:15:46 (47) (326)
response 0123 ;
```

Simple command message: time

SMS command message:

```
execute 0124 time;
```

SMS response message:

```
HAKSYS CLOGGER V1.00 28D49259 000002 22/11/2010 06:18:06 (47) (326)
response 0124 06:18:06;
```

Multiple commands message: beep and time

SMS command message:

```
execute 0210 beep; time;
```

SMS response message:

```
HAKSYS CLOGGER V1.00 28D49259 000002 22/11/2010 06:19:52 (47) (326)
response 0210 ;06:19:52;
```

Simple command message: beep

SMS command message:

exec 0123 beep;

SMS response message:

response 0123 ;

Simple command message: time

SMS command message:

exec 0124 time;

SMS response message:

response 0124 09:30:38 ;

Multiple commands message: beep and time

SMS command message:

exec 0210 beep; time;

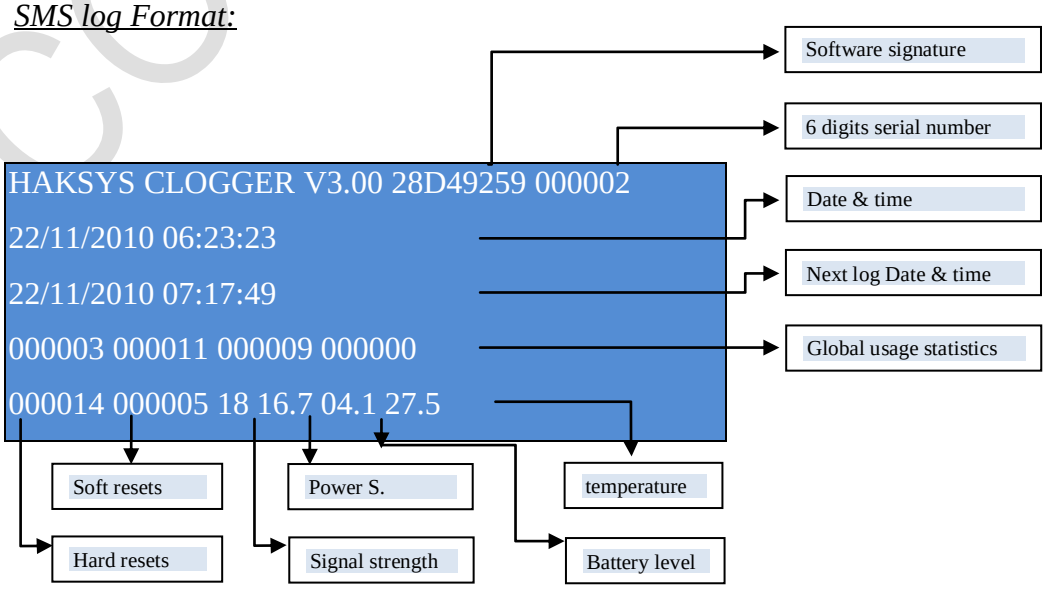
SMS response message:

response 0210 ;09:30:38 ;

The SMS Log

The *CLOGGER* can be configured to send periodically an SMS message to the monitoring center. This message contains status and usage statistics.

SMS log Format:



The SMS Alarm

The *CLOGGER* can be configured to send SMS alarms to the monitoring center. These alarms include error notifications and low supplies (paper, toner or drum) warnings .

Commands description

Note: refer to "Commands set summary" section to get an overview and a brief description.
Note: all the examples presented in this section are using a console connection for the sake of simplicity.

The commands are organized in 8 categories:

- System
- Memory
- Wireless
- Crypto
- Input/output
- History
- General
- Debug

System commands:

Version: *(version, cmd0002)*

Key word	*Parameters*
cmd0002	*None*

Get:

```
$cmd0002 ↵
V3.00
$
```

Echo: *(echo, cmd0003)*

Key word	*Parameters*
cmd0003	*A string of ascii characters*

Echo "hello world"

```
$cmd0003 hello world ↵
hello world
$
```

Reset: *(reset, cmd0004)*

Key word	*Parameters*
cmd0004	*None*

Execute:

```
$ cmd0004 ↵
restarting system
$
```

Beep: *(beep, cmd0005)*

Key word	*Parameter 1*	*Parameter 2*
cmd0005	*Number of beeps (opt.)*	*Beep duration in milli seconds (opt.)*

Execute:

```
$ cmd0005 ↵
$
```

Make 3 beeps of 70 ms:

```
$ cmd0005 3 70 ↵
$
```

Power supply (Vin): *(volt, cmd0006)*

Key word	*Parameters*
cmd0006	*None*

Get:

```
$ cmd0006 ↵
24.1V
$
```

Temperature: *(temp, cmd0007)*

Key word	*Parameters*
cmd0007	*None*

Get:

```
$ cmd0007 ↵
23.5C
$
```

Time: *(time, cmd0008)*

Key word	*Parameter 1*	*Parameter 2*	*Parameter 3*
cmd0008	*Hours (opt.)*	*Minutes (opt.)*	*Seconds (opt.)*

Get time:

```
$ cmd0008 ↵
10:19:06
$
```

Set time to 14:32:28:

```
$ cmd0008 14 32 28 ↵
14:32:28
$
```

Date: *(date, cmd0009)*

Key word	*Parameter 1*	*Parameter 2*	*Parameter 3*
cmd0009	*Day (opt.)*	*Month (opt.)*	*Year (opt.)*

Get date:

```
$ cmd0009 ↵
23/11/2010
$
```

Set date to 26 Nov. 2010:

```
$ cmd0009 26 11 2010 ↵
26/11/2010
$
```

Date & time: *(datetime, cmd0010)*

Key word	*Param. 1*	*Param. 2*	*Param. 3*	*Param. 4*	*Param. 5*	*Param. 6*
Cmd0010	*Day (opt.)*	*Month (opt.)*	*Year (opt.)*	*Hour (opt.)*	*Minute (opt.)*	*Second (opt.)*

Get date & time:

```
$ cmd0010 ↵
30/06/2011 04:30:26
$
```

Set date & time to 21 Oct. 2010 16:57:00

```
$ cmd0010 21 10 2010 16 57 00 ↵
21/10/2010 16:57:00
$
```

Application: *(appli, cmd0011)*

Key word	*Parameters*
cmd0011	*None*

Get:

```
$cmd0011 ↵
HAKSYS CLOGGER
$
```

Application: *(build, cmd0012)*

Key word	*Parameters*
cmd0012	*None*

Get:

```
$cmd0012 ↵
HAKSYS CLOGGER
$
```

Signature: *(sign, cmd0013)*

Key word	*Parameters*
cmd0013	*None*

Get:

```
$cmd0013 ↵
BE78095A
$
```

Memory commands:

Verify Memory integrity : *(mci, cmd0020)*

Key word	*Parameters*
Cmd0020	*None*

Execute:

Response: "0" for success

```
$cmd0020 ↵
0
$
```

Reset all memory: *(mrall, cmd0021)*

Key word	*Parameters*
Cmd0021	*None*

```
$cmd0021 ↵
$
```

Set all memory: *(msall, cmd0022)*

Key word	*Parameters*
Cmd0022	*None*

```
$cmd0022 ↵
$
```

Serial number: *(mpsrn, cmd0040)*

Key word	*Parameter 1*
Cmd0040	*A 6 digits number*

Get:

```
$cmd0040 ↵
000002
$
```

Set serial to 233:

```
$cmd0040 233 ↵
$
```

Groups: *(mpg, cmd0041)*

Key word	*Parameter 1*	*Parameter 2*
Cmd0041	*Group index [1-10] (opt.)*	*Group (opt.)*

Get groups:

```
$cmd0041 ↵
0 0 0 0 0 0 0 0 0 0
$
```

Get group at index 1:

```
$cmd0041 1
0
$
```

Set group at index 1 to 15:

```
$ cmd0041 1 15
$
```

Set group at index 4 to 24:

```
$ cmd0041 4 24
$
```

Get groups:

```
$ cmd0041
15 0 0 24 0 0 0 0 0 0
$
```

Get group at index 1:

```
$ cmd0041 1
15
$
```

Get group at index 4:

```
$ cmd0041 4
24
$
```

Filter: *(mpf, cmd0042)*

Key word	*Parameter 1*
Cmd0042	*Samples in milli second*

Get:

```
$cmd0042
20
$
```

Set filter to 40:

```
$cmd0042 40
$
```

Admin. code: *(mpac, cmd0043)*

Key word	*Parameter 1*
Cmd0043	*A 6 digits number*

Get:

```
$cmd0043
000000
$
```

Set admin. code to 1234:

```
$ cmd0043 1234
$
```

Get:

```
$ cmd0043
001234
$
```

Team. code: *(mptc, cmd0044)*

Key word	*Parameter 1*
Cmd0044	*A 6 digits number*

Get:

```
$cmd0044 ↵
000000
$
```

Set team code to 6789:

```
$ cmd0044 6789 ↵
$
```

Beep: *(mpb, cmd0045)*

Key word	*Parameter 1*
Cmd0045	*A Boolean (0 or 1)*

Get:

```
$cmd0045 ↵
0
$
```

Set beep:

```
$ cmd0045 1 ↵
$
```

Optos mode: *(mpom, cmd0046)*

Key word	*Parameter 1*
Cmd0046	*Mode [0-23]*

Get:

```
$ cmd0046 ↵
0
$
```

Set optos mode 13:

```
$ cmd0046 13 ↵
$
```

Relays mode: *(mprm, cmd0047)*

Key word	*Parameter 1*
Cmd0047	*Mode [0-3]*

Get:

```
$ cmd0047 ↵
0
$
```

Set relays mode 3:

```
$ cmd0047 3 ↵
$
```

RF Band: *(mprfb, cmd0048)*

Key word	*Parameter 1*
Cmd0048	*Mode [0-2]*

Get:

```
$ cmd0048 ↵
0
$
```

Set Rf band 2:

```
$ cmd0048 2 ↵
$
```

RF Data Rate: *(mprfdr, cmd0049)*

Key word	*Parameter 1*
Cmd0049	*Mode [0-6]*

Get:

```
$ cmd0049 ↵
0
$
```

Set Rf data rate 2:

```
$ cmd0049 2 ↵
$
```

RF Channel: *(mprfc, cmd0050)*

Key word	*Parameter 1*
Cmd0050	*Mode [0-6]*

Get:

```
$ cmd0050 ↵
0
$
```

Set rf channel rate 1:

```
$ cmd0050 1 ↵
$
```

Sms Log Period: *(mplsp, cmd0051)*

Key word	*Parameter 1*	*Parameter 2*	*Parameter 3*	*Parameter 4*	*Parameter 5*
Cmd0051	*Months*	*weeks*	*days*	*hours*	*Minutes*

Get:

```
$ cmd0051 ↵
00  00 00 01 00
$
```

Set period to 5 minutes:

```
$ cmd0051 0 0 0 0 5 ↵
$
```

Set period to 3 weeks:

```
$ cmd0051 0 3 0 0 0 ↵
$
```

Sms Log starting date & time: *(mplsst, cmd0052)*

Key word	*Param. 1*	*Param. 2*	*Param. 3*	*Param. 4*	*Param. 5*	*Param. 6*
Cmd0052	*Day (opt.)*	*Month (opt.)*	*Year (opt.)*	*Hour (opt.)*	*Minute (opt.)*	*Second (opt.)*

Get:

```
$ cmd0052 ↵
21/11/2010 00:00:00
$
```

Set :

21 Nov. 2010 01:00:00

```
$ cmd0052  21 11 2010 01 00 00 ↵
21/11/ 2010 01:00:00
$
```

Sms Log ending date & time: *(mplset, cmd0053)*

Key word	*Param. 1*	*Param. 2*	*Param. 3*	*Param. 4*	*Param. 5*	*Param. 6*
Cmd0053	*Day (opt.)*	*Month (opt.)*	*Year (opt.)*	*Hour (opt.)*	*Minute (opt.)*	*Second (opt.)*

Get:

```
$ cmd0053 ↵
21/11/2010 00:00:00
$
```

Set: 31 Dec. 2011 00:00:00

```
$ cmd0053 31 12 2011 00 00 00 ↵
31/12/ 2011 00:00:00
$
```

Sms Header: *(mpsh, cmd0062)*

Key word	*Parameter 1*
Cmd0062	*1 to 16 characters (opt.)*

Get:

```
$cmd0062 ↵
COPY:
$
```

Set header to vm2m:

```
$ cmd0062  vm2m ↵
$
```

Func. mode: *(mpfm, cmd0063)*

Key word	*Parameter 1*
Cmd0063	*A number (0 or 3)*

Get:

```
$cmd0063 ↵
0
$
```

Set beep:

```
$ cmd0063 1 ↵
$
```

Price1: *(mpp1, cmd0064)*

Key word	*Parameter 1*
Cmd0064	*A 6 digits number*

Get:

```
$cmd0064 ↵
10
$
```

Set price1 to 20 cents:

```
$cmd0064 20 ↵
$
```

Price2: *(mpp2, cmd0065)*

Key word	*Parameter 1*
Cmd0065	*A 6 digits number*

Get:

```
$cmd0065 ↵
10
$
```

Set price2 to 20 cents:

```
$cmd0065 20 ↵
$
```

Price3: *(mpp3, cmd0066)*

Key word	*Parameter 1*
Cmd0066	*A 6 digits number*

Get:

```
$cmd0066 ↵
10
$
```

Set price3 to 20 cents:

```
$cmd0066 20 ↵
$
```

Price4: *(mpp4, cmd0067)*

Key word	*Parameter 1*
Cmd0067	*A 6 digits number*

Get:

```
$cmd0067 ↵
10
$
```

Set price4 to 20 cents:

```
$cmd0067 20 ↵
$
```

Interim Price1: *(mppi1, cmd0069)*

Key word	*Parameter 1*
Cmd0069	*A 6 digits number*

Get:

```
$cmd0069 ↵
10
$
```

Set interim price1 to 20 cents:

```
$cmd0069 20 ↵
$
```

Interim Price2: *(mppi2, cmd0070)*

Key word	*Parameter 1*
Cmd0070	*A 6 digits number*

Get:

```
$cmd0070 ↵
10
$
```

Set interim price2 to 20 cents:

```
$cmd0070  20 ↵
$
```

Interim Price3: *(mppi3, cmd0071)*

Key word	*Parameter 1*
Cmd0071	*A 6 digits number*

Get:

```
$cmd0071 ↵
10
$
```

Set interim price3 to 20 cents:

```
$cmd0071  20 ↵
$
```

Interim Price4: *(mppi4, cmd0072)*

Key word	*Parameter 1*
Cmd0072	*A 6 digits number*

Get:

```
$cmd0072 ↵
10
$
```

Set interim price4 to 20 cents:

```
$cmd0072  20 ↵
$
```

Global counter A4/BW: *(msf1, cmd0080)*

Key word	*Parameter 1*
Cmd0080	*New counter value (opt)*

Get:

```
$cmd0080 ↵
2
$
```

Set counter to 0:

```
$cmd0080 0 ↵
$
```

Global counter A3/BW: *(msf2, cmd0081)*

Key word	*Parameter 1*
Cmd0081	*New counter value(opt)*

Get:

```
$cmd0081 ↵
2
$
```

Set counter to 1000:

```
$cmd0081 1000 ↵
$
```

Global counter A4/COLOR: *(msf3, cmd0082)*

Key word	*Parameter 1*
Cmd0082	*New counter value(opt)*

Get:

```
$cmd0082 ↵
2
$
```

Set counter to 2000:

```
$cmd0082 2000 ↵
$
```

Global counter A3/COLOR: *(msf4, cmd0083)*

Key word	*Parameter 1*
Cmd0083	*New counter value(opt)*

Get:

```
$cmd0083 ↵
2
$
```

Set counter to 3000:

```
$cmd0083 3000 ↵
$
```

Usage statistics: *(msfall, cmd0084)*

Key word	*Parameter 1*
Cmd0084	*none*

Get:

```
$cmd0084 ↵
0 1000 2000 3000
$
```

Interim counter A4/BW: *(msif1, cmd0085)*

Key word	*Parameter 1*
Cmd0085	*New counter value (opt)*

Get:

```
$cmd0085 ↵
2
$
```

Set counter to 0:

```
$cmd0080 5 ↵
$
```

Interim counter A3/BW: *(msif2, cmd0086)*

Key word	*Parameter 1*
Cmd0086	*New counter value(opt)*

Get:

```
$cmd0086 ↵
2
```

Set counter to 1000:

```
$cmd0086 1000 ↵
$
```

Interim counter A4/COLOR: *(msif3, cmd0087)*

Key word	*Parameter 1*
Cmd0087	*New counter value(opt)*

Get:

```
$cmd0087 ↵
2
$
```

Set counter to 2000:

```
$cmd0087 2000 ↵
$
```

Interim counter A3/COLOR: *(msif4, cmd0088)*

Key word	*Parameter 1*
Cmd0088	*New counter value(opt)*

Get:

```
$cmd0088 ↵
2
$
```

Set counter to 3000:

```
$cmd0088 3000 ↵
$
```

Interim statistics: *(msifall, cmd0089)*

Key word	*Parameter 1*
Cmd0089	*none*

Get:

```
$cmd0089 ↵
0 1000 2000 3000
$
```

Work counter A4/BW: *(mswf1, cmd0090)*

Key word	*Parameter 1*
Cmd0090	*New counter value (opt)*

Get:

```
$cmd0090 ↵
2
$
```

Set counter to 0:

```
$cmd0090 5 ↵
$
```

Work counter A3/BW: *(mswf2, cmd0091)*

Key word	*Parameter 1*
Cmd0091	*New counter value(opt)*

Get:

```
$cmd0091 ↵
2
$
```

Set counter to 1000:

```
$cmd0091 1000 ↵
$
```

Work counter A4/COLOR: *(mswf3, cmd0092)*

Key word	*Parameter 1*
Cmd0092	*New counter value(opt)*

Get:

```
$cmd0092 ↵
2
$
```

Set counter to 2000:

```
$cmd0092 2000 ↵
$
```

Work counter A3/COLOR: *(mswf4, cmd0093)*

Key word	*Parameter 1*
Cmd0093	*New counter value(opt)*

Get:

```
$cmd0093 ↵
2
$
```

Set counter to 3000:

```
$cmd0093 3000 ↵
$
```

Work statistics: *(mswall, cmd00944)*

Key word	*Parameter 1*
Cmd0094	*none*

Get:

```
$cmd0094 ↵
0 1000 2000 3000
$
```

Credit: *(msc, cmd0095)*

Key word	*Parameter 1*
Cmd0095	*New value(opt)*

Get:

```
$cmd0095 ↵
2
$
```

Set counter to 2000:

```
$cmd0095 2000 ↵
$
```

Turnover: *(mst, cmd0096)*

Key word	*Parameter 1*
Cmd0096	*New value(opt)*

Get:

```
$cmd0096 ↵
2
$
```

Set counter to 2000:

```
$cmd0096 2000 ↵
$
```

Hardware resets counter: *(mchr, cmd0100)*

Key word	*Parameter 1*
Cmd0100	*New counter value*

Get:

```
$cmd0100 ↵
2
$
```

Set counter to 100:

```
$cmd0100 100 ↵
$
```

Software resets counter: *(mcsr, cmd0101)*

Key word	*Parameter 1*
Cmd0101	*New counter value*

Get:

```
$cmd0101 ↵
2
$
```

Set counter to 200:

```
$cmd0101 200 ↵
$
```

Memory errors counter: *(mcee, cmd0102)*

Key word	*Parameter 1*
Cmd0102	*New counter value*

Get:

```
$cmd0102 ↵
2
$
```

Set counter to 200:

```
$cmd0102 200 ↵
$
```

SMS sent counter: *(mcsmss, cmd0103)*

Key word	*Parameter 1*
Cmd0103	*New counter value*

Get:

```
$cmd0103 ↵
2
$
```

Set counter to 200:

```
$cmd0103 200 ↵
$
```

SMS sent with success counter : *(mcsmsss, cmd0104)*

Key word	*Parameter 1*
Cmd0104	*New counter value*

Get:

```
$cmd0104 ↵
2
$
```

Set counter to 200:

```
$cmd0104 200 ↵
$
```

SMS sent with failure counter: *(mcsmssf, cmd0105)*

Key word	*Parameter 1*
Cmd0105	*New counter value*

Get:

```
$cmd0105 ↵
2
$
```

Set counter to 200:

```
$cmd0105 200 ↵
$
```

SMS received counter: *(mcsmsr, cmd0106)*

Key word	*Parameter 1*
Cmd0106	*New counter value*

Get:

```
$cmd0106 ↵
2
$
```

Set counter to 200:

```
$cmd0106 200 ↵
$
```

Power on date & time: *(mcpont, cmd0107)*

Key word	*Parameters*
Cmd0107	*none*

Get:

```
$cmd0107 ↵
24/11/ 2010 07:18:26
$
```

Power off date & time: *(mcpofft, cmd0108)*

Key word	*Parameters*
Cmd0108	*none*

Get:

```
$cmd0108 ↵
Thursday Jan 01 1970 00:00:00 (01) (001)
$
```

Log SMS sent counter: *(mclsmss, c*

Key word	*Parameter 1*
Cmd0109	*New counter value*

Get:

```
$cmd0109 ↵
7
$
```

Set counter to 0:

```
$cmd0109 0 ↵
$
```

Log SMS sent with success counter : *(mclsmsss, cmd0110)*

Key word	*Parameter 1*
Cmd0110	*New counter value*

Get:

```
$cmd0110 ↵
7
$
```

Set counter to 0:

```
$cmd011 0 ↵
$
```

Log SMS sent with failure counter: *(mclsmssf, cmd0111)*

Key word	*Parameter 1*
Cmd0111	*New counter value*

Get:

```
$cmd0111 ↵
5
$
```

Set counter to 0:

```
$cmd0111  0 ↵
$
```

Last logged SMS date & time: *(mclsmslt, cmd0112)*

Key word	*Parameters*
Cmd0112	*none*

Get:

```
$cmd0112 ↵
24/11/ 2010 01:18:26
$
```

Next logged SMS date & time: *(mclsmslt, cmd0113)*

Key word	*Parameters*
Cmd0113	*none*

Get:

```
$cmd0113 ↵
24/11/ 2010 06:18:26
$
```

Factory default parameters: *(mfdp, cmd0030)*

Key word	*Parameters*
Cmd0030	*none*

Execute:

```
$cmd0030 ↵
$
```

Factory default usage statistics: *(mfds, cmd0031)*

Key word	*Parameters*
Cmd0031	*none*

Execute:

```
$cmd0031 ↵
$
```

Factory default counters: *(mfdc, cmd0032)*

Key word	*Parameters*
Cmd0032	*none*

Execute:

```
$cmd0032 ↵
$
```

Factory default incoming Sms (history): *(mfdish, cmd0033)*

Key word	*Parameters*
Cmd0033	*None*

Execute:

```
$cmd0033 ↵
$
```

Factory default outgoing Sms (history): *(mfdosh, cmd0034)*

Key word	*Parameters*
Cmd0034	*None*

Execute:

```
$cmd0034 ↵
$
```

Factory default incoming calls (history): *(mfdich, cmd0035)*

Key word	*Parameters*
Cmd0035	*None*

Execute:

```
$cmd0035 ↵
$
```

Factory default outgoing calls (history): *(mfdoch, cmd0036)*

Key word	*Parameters*
Cmd0036	*None*

Execute:

```
$cmd0036 ↵
$
```

Factory default all: *(mfdall, cmd0037)*

Key word	*Parameters*
Cmd0037	*None*

Execute:

```
$cmd0037 ↵
$
```

Wireless commands:

Sim pin code: *(mmpin, cmd0054)*

Key word	*Parameter 1*
Cmd0054	*4 to 8 digits number (opt.)*

Get:

```
$cmd0054 ↵
$
```

Set pin to 012345:

```
$ cmd0054 012345 ↵
$
```

Get:

Set pin to empty:

```
$ cmd0054 ↵
012345
$ ↵
```

Get:

```
$ cmd0054 empty
$ cmd0054 ↵
$
```

Sim puk code: *(mmpuk, cmd0055)*

Key word	*Parameter 1*
Cmd0055	*4 to 8 digits number (opt.)*

Get:

```
$cmd0055 ↵
$
```

Set puk to 543210:

```
$ cmd0055 543210 ↵
$
```

Get:

```
$ cmd0055 ↵
543210
$
```

Set puk to empty:

```
$ cmd0055 empty ↵
$
```

Get:

```
$ cmd0055 ↵
$
```

Subscriber phone number: *(mmsp, cmd0056)*

Key word	*Parameter 1*
Cmd0056	*<+> 15 digits number (opt.)*

Get:

```
$cmd0056 ↵
$
```

Set phone to +33626200235:

```
$ cmd0056 +33626200235 ↵
$
```

Get:

```
$ cmd0056 ↵
+33626200235
$
```

Set phone to empty:

```
$ cmd0056 empty ↵
$
```

Get:

```
$ cmd0056 ↵
$
```

Monitoring center phone number 1: *(mmmcp1, cmd0057)*

Key word	*Parameter 1*
Cmd0057	*<+> 15 digits number (opt.)*

Get:

```
$cmd0057 ↵
$
```

Set phone to +33626200235:

```
$ cmd0057 +33626200235 ↵
$
```

Get:

```
$ cmd0057 ↵
+33626200235
$
```

Set phone to empty:

```
$ cmd0057 empty ↵
$
```

Get:

```
$ cmd0057 ↵
$
```

Monitoring center phone number 2: *(mmmcp2, cmd0058)*

Key word	*Parameter 1*
Cmd0058	*<+> 15 digits number (opt.)*

Get:

```
$cmd0058 ↵
$
```

Set phone to +33626200235:

```
$ cmd0058 +33626200235 ↵
$
```

↵

Get:

```
$ cmd0058
+33626200235
$
```

Set phone to empty:

```
$ cmd0058  empty ↵
$
```

Get:

```
$ cmd0058 ↵
$
```

Servicing center phone number 1: *(mmscp1, cmd0059)*

Key word	*Parameter 1*
Cmd0059	*<+> 15 digits number (opt.)*

Get:

```
$cmd0059 ↵
$
```

Set phone to +33626200235:

```
$ cmd0059  +33626200235 ↵
e
$ cmd0059 ↵
```

Get:

```
+33626200235
$
```

Set phone to empty:

```
$ cmd0059  empty ↵
$
```

Get:

```
$ cmd0059 ↵
$
```

Servicing center phone number 2: *(mmscp2, cmd0060)*

Key word	*Parameter 1*
Cmd0060	*<+> 15 digits number (opt.)*

Get:

```
$cmd0060 ↵
$
```

Set phone to +33626200235:

```
$ cmd0060  +33626200235 ↵
$
```

Get:

```
$ cmd0060 ↵
+33626200235
$
```

Set phone to empty:

```
$ cmd0060 empty ↵
$
```

Get:

```
$ cmd0060 ↵
$
```

Send SMS message: *(smssm, cmd0120)*

Key word	*Parameter 1*	*Parameter 2*
Cmd0120	*Phone number*	*Message*

Send Sms to:0626200235
"M2M Connecting machines"
Response: "0" for success

```
$ cmd0120  0626200235 M2M Connecting machines
0
$
```

Count Sms messages (SIM): *(smscm, cmd0121)*

Key word	*Parameter s*
Cmd0121	*None*

Get:

```
$cmd0121
4
$
```

Read Sms message (SIM): *(smsrm, cmd0122)*

Key word	*Parameter 1*
Cmd0122	*Location [1-30]*

Read message at location 1:

```
$cmd0122 1
+33626200235
M2M Connecting machines
$
```

Read Sms message in brute format (SIM): *(smsrmb, cmd0123)*

Key word	*Parameter 1*
Cmd0123	*Location [1-30]*

Read message at location 1:

```
$cmd0123 1
"REC READ","+33626200235","Abdelhak","10/11/21,20:18:59+04"
M2M Connecting machines
$
```

Read all Sms messages (SIM): *(smsrmall, cmd0124)*

Key word	*Parameters*
Cmd0124	*None*

Execute:

```
$cmd0124
Total number of messages is 1
#1
+33626200235
M2M Connecting machines
$
```

Read all Sms messages brute (SIM): *(smsrmball, cmd0125)*

Key word	*Parameters*
Cmd0125	*None*

Get:

```
$cmd0125
Total number of messages is 1

#1
"REC READ","+33626200235","Abdelhak","10/11/21,20:18:59+04"M2M
Connecting machines
$
```

Delete Sms message (SIM): *(smsdm, cmd0126)*

Key word	*Parameter 1*
Cmd0126	*Location [1-30]*

Delete message at location 1:

```
$cmd0126 1
0
$
```

Delete all Sms messages (SIM): *(smsdmall, cmd0127)*

Key word	*Parameter 1*
Cmd0127	*None*

Execute:

```
$cmd0127
0
$
```

Sms Get Log: *(smsgl, cmd0128)*

Key word	*Parameter 1*
Cmd0128	*none*

Get log:

```
$cmd0128
COPY: HAKSYS CLOGGER V3.00 4361242B 000101
30/06/2011 06:44:31
01/07/2011 04:58:36
000000 000000 000000 000000
000002 000000 12 24.0 04.1 30.0
$
```

Sms Get Log RF: *(smsglrf, cmd0129)*

Key word	*Parameter 1*
Cmd0129	*Group*

Get log RF 106:

```
$cmd0129 106
COPY: HAKSYS CLOGGER V3.00 4361242B 000106
30/06/2011 07:44:31
01/07/2011 07:58:36
000000 000000 000000 000000
000002 000000 13 24.0 04.1 30.0
$
```

Sms Get Log All: *(smsglall, cmd0130)*

Key word	*Parameter 1*
Cmd0129	*none*

Get log all:

```
$cmd0130
COPY: HAKSYS CLOGGER V3.00 4361242B 000106
30/06/2011 07:44:31
01/07/2011 07:58:36
000000 000000 000000 000000
000002 000000 13 24.0 04.1 30.0
$
```

Send Sms Log: *(smssl, cmd0131)*

Key word	*Parameter*
Cmd318	*Phone Num*

Send Sms log to 0626200235:

```
$cmd318  0626200235
142
$
```

Send Sms Log RF: *(smsglrf, cmd0132)*

Key word	*Parameter 1*	*Parameter 2*
Cmd0132	*Phone number*	*Group*

Send Sms log to 0626200235:

```
$cmd0132  0626200235 106
142
$
```

Send Sms Log All: *(smsslall, cmd0133)*

Key word	*Parameter 1*
Cmd0133	*Phone Number*

Send Sms log to 0626200235:

```
$cmd0133  0626200235
142
$
```

Call phone number: *(mdial, cmd0140)*

Key word	*Parameter 1*
Cmd0140	*Phone Number*

Dial 0954009306:

```
$cmd0140  0626200235
0
$
```

Hang a call: *(mhang, cmd0141)*

Key word	*Parameter s*
Cmd0141	*None*

Hang:

```
$cmd0141
0
$
```

Answer a call: *(manswer, cmd0142)*

Key word	*Parameter s*
Cmd331	*None*

Answer:

```
$cmd0142 ↵
0
$
```

Sms service center: *(mssca, cmd0150)*

Key word	*Parameters*
Cmd0150	*None*

Get:

```
$cmd0150 ↵
+33609001390
$
```

Modem serial number identification: *(msni, cmd0151)*

Key word	*Parameters*
Cmd0151	*None*

Get:

```
$cmd0151 ↵
012207000122595
$
```

Modem signal strength: *(mcsq, cmd0152)*

Key word	*Parameters*
Cmd0152	*None*

Get:

```
$cmd0152 ↵
19
$
```

Modem temperature: *(mtemp, cmd0153)*

Key word	*Parameters*
Cmd0153	*None*

Get:

```
$cmd0153 ↵
28.0C
$
```

Modem battery: *(mbatt, cmd0154)*

Key word	*Parameters*
Cmd0154	*None*

Get:

```
$cmd0154 ↵
04.1V
$
```

Search for a wireless channel: *(rfsch, cmd0160)*

Key word	*Parameters*
Cmd0160	*None*

Get:

```
$cmd0160 ↵
6
$
```

wireless configuration: *(rfconfig, cmd0161)*

Key word	*Parameters*
Cmd0161	*None*

Get:

```
$cmd0161 ↵
BAND[868]
DATARATE[115200]
CHANNELS[7]
CH[0]
$
```

Crypto:

Check crypto memory: *(crptc, cmd0170)*

Key word	*Parameters*
Cmd0170	*None*

Get:
Response:
"0" for success
Other for failure

```
$cmd0170 ↵
0
$
```

Crypto serial number: *(crpts, cmd0171)*

Key word	*Parameters*
Cmd0171	*None*

Get:

```
$cmd0171 ↵
2
$
```

Crypto identity number: *(crptid, cmd0172)*

Key word	*Parameters*
Cmd0172	*None*

Get:

```
$cmd0172 ↵
17102010183655
$
```

Input/output:

Relay1 i/o: *(iorly1, cmd0180)*

Key word	*Parameter 1*
Cmd0180	*New state*

Get status:

```
$cmd0180 ↵
0
$
```

Set relay1:

```
$cmd0180 1 ↵
$
```

Reset relay1:

```
$cmd0180 0 ↵
$
```

Relay2 i/o: *(iorly2, cmd0181)*

Key word	*Parameter 1*
Cmd0181	*New state*

Get status:

```
$cmd181 ↵
0
$
```

Set relay2:

```
$ cmd181 1 ↵
$
```

Reset relay2:

```
$ cmd181 0 ↵
$
```

Optos i/o: *(ioop, cmd0182)*

Key word	*Parameters*
Cmd0182	*None*

Get:

```
$cmd0182 ↵
000
$
```

Opto pulse i/o: *(ioop, cmd0183)*

Key word	*Parameters*
Cmd0183	*None*

Get:

```
$cmd0183 ↵
0
$
```

Opto format i/o: *(iofmt, cmd0184)*

Key word	*Parameters*
Cmd0184	*None*

Get:

```
$cmd0184 ↵
0
$
```

Opto color i/o: *(iocol, cmd0185)*

Key word	*Parameters*
Cmd0185	*None*

Get:

```
$cmd0185 ↵
0
$
```

Modem reset signal i/o: *(iomrst, cmd0186)*

Key word	*Parameter 1*
Cmd0186	*New state*

Get:

```
$cmd0186 ↵
0
$
```

Set:

```
$cmd0186 1 ↵
$
```

Modem pwrhey signal i/o: *(iompk, cmd0187)*

Key word	*Parameter 1*
Cmd0187	*New state*

Get:

```
$cmd0187 ↵
0
$
```

Set:

```
$cmd0187 1 ↵
$
```

Modem dtr signal i/o: *(iomdtr, cmd0188)*

Key word	*Parameter 1*
Cmd0188	*New state*

Get:

```
$cmd0188 ↵
0
$
```

Set:

```
$cmd608 1 ↵
$
```

History:

List incoming calls: *(mhic, cmd0190)*

Key word	*Parameters*
Cmd0190	*None*

Get:

```
$cmd0190 ↵
Total incoming calls is 5
  [#05 ] 24/11/2010 18:42:35 (47) (328) 0954009306
  [#04 ] 24/11/2010 18:39:47 (47) (328) 0954009306
  [#03 ] 21/11/2010 11:05:43 (46) (325) 0663752237
  [#02 ] 21/11/2010 11:04:51 (46) (325) 0663752237
  [#01 ] 21/11/2010 09:58:02 (46) (325) 0954009306
$
```

List outgoing calls: *(mhoc, cmd0191)*

Key word	*Parameters*
Cmd0191	*None*

Get:

```
$cmd0191 ↵
Total outgoing calls is 0
$
```

List incoming Sms: *(mhis, cmd0192)*

Key word	*Parameters*
Cmd0192	*None*

Get:

```
$cmd0192 ↵
Total incoming Sms is 1
  [#01 ] 24/11/2010 07:25:44 (47) (328)
  "REC UNREAD","+33626200235","Abdelhak","10/11/24,07:26:22+04"
HAKSYS CLOGGER V1.00 A3BEA618 000002
24/11/2010 07:25:36 (47) (328)
27/11/2010 14:54:01 (47) (331)
007000 001000 002000 003000
000102 000000 16 16.7 04.1 27.5
$
```

List outgoing Sms: *(mhos, cmd0193)*

Key word	*Parameters*
Cmd0193	*None*

Get:

```
$cmd0193 ↵
Total Outgoing sms is 1
  [#01 ] 22/11/2010 03:17:50 (47) (326)
 0626200235 HAKSYS CLOGGER V1.00 28D49259 000002
22/11/2010 03:17:50 (47) (326)
22/11/2010 04:17:49 (47) (326)
000003 000011 000009 000000
000014 000005 16 16.7 04.1 27.0
$
```

General:

Application settings Check: *(acs, cmd0200)*

Key word	*Parameters*
Cmd0200	*None*

Check:

```
$cmd0200
check calendar ...OK

 check crypto ...OK

 check memory ...OK

 check param serial ...OK

 check params log...OK

 check params modem...OK
$
```

Application parameters Info.: *(aip, cmd0201)*

Key word	*Parameters*
Cmd0201	*None*

Get:

```
$cmd0201
wear:                   0
 serial:                101
 groups:                0 0 0 0 0 0 0 0 0 0
 filter:                20
 cadmin:               0
 cteam:                0
 beep:                   1
 relays mode:          3
 optos_mode:           0
 rf_band:               0
 rf_datarate:          0
 rf_channel:           0
 log sms period:       00 00 01 00 00
 log staring time:      30/06/2011 04:58:36
 log ending time:       29/06/2012 04:58:36
 pin code:             0000
 puk code:             0000
 phone subscriber :          0000000000
 phone info center1:         0626514496
 phone info center2:         0000000000
 phone maintenance center1:    0626200235
 phone maintenance center2:    0000000000
 dbg_level:             0
 log sms header:       COPY:
$
```

Application counters Info.: *(aic, cmd0202)*

Key word	*Parameters*
Cmd0202	*None*

Get:

```
$cmd0202
wear:                32
 hard resets:        2
 soft resets:        0
 eep err:            0
 sms sent:           7
 sms sent success:   7
 sms sent failure:   0
 sms received:       4
 power on time       30/06/2011 06:33:42
 power off time      01/01/1970 00:00:00
 log sms sent:       3
 log sms sent success:  3
 log sms sent failure:  0
 log sms last time:     30/06/2011 06:50:47
 log sms next time:     01/07/2011 06:50:47
$
```

Debug:

Shell mode: *(shm, cmd0900)*

Key word	*Parameters*
Cmd0900	*None*

Get:

```
$cmd0900
0
$
```

Set:

```
$cmd0900 1
$
```

Trace default procrdures flag: *(tdpf, cmd0901)*

Key word	*Parameters*
Cmd0901	*None*

Get:

```
$cmd0901
0
$
```

Set:

```
$cmd0901 1
$
```

Trace events flag: *(tef, cmd0902)*

Key word	*Parameters*
Cmd0902	*None*

Get:

```
$cmd0902 ↵
0
$
```

Set:

```
$cmd0902 1 ↵
$
```

Trace idle tasks flag: *(titf, cmd0903)*

Key word	*Parameters*
Cmd0903	*None*

Get:

```
$cmd0903 ↵
0
$
```

Set:

```
$cmd0903 1 ↵
$
```

Commands set summary

1. **System**

Command	*Alias*	*Parameters*	*Description*
help	*cmd001*	*No*	*Print commands summary table*
version	*cmd002*	*No*	*software version*
echo	*Cmd003*	*Mandatory*	*Echo the string parameter itself*
reset	*Cmd004*	*No*	*System software reset*
beep	*Cmd005*	*Optional*	*Buzzer beep*
volt	*Cmd006*	*No*	*System power supply "Vin"*
temp	*Cm007*	*No*	*System temperature*
time	*Cmd008*	*Optional*	*System time*
date	*Cmd009*	*Optional*	*System date*
datetime	*Cmd010*	*Optional*	*System date & time*
Appli	*Cmd011*	*No*	*Application name*
Build	*Cmd012*	*No*	*Software compilation date*
Sign	*Cmd013*	*No*	*Firmware signature*

2. **Memory**

a. *general*

Command	*Alias*	*Parameters*	*Description*
mci	*Cmd0020*	*No*	*Memory integrity sanity check*
mrall	*Cmd0021*	*No*	*Reset all memory to "x00"*
msall	*Cmd0022*	*No*	*Reset all memory to "xFF"*

b. *Parameters*

Command	*Alias*	*Parameters*	*Description*
mpsrn	*Cmd0040*	*No*	*System serial number*
mpg	*Cmd0041*	*Optional*	*System groups*
mpf	*Cmd0042*	*optional*	*Copier pulse filter (ms)*
mpca	*Cmd0043*	*optional*	*System administration code*
mpct	*Cmd0044*	*optional*	*System team code*
mpb	*Cmd0045*	*optional*	*Enables or disable system beep*
mpom	*Cmd0046*	*optional*	*Optos configuration*
mprm	*Cmd0047*	*optional*	*Relays configuration*
mprfb	*cmd0048*	*optional*	*Rf band*
mprfdr	*cmd0049*	*optional*	*Rf data rate*
mprfc	*cmd0050*	*optional*	*Rf channel*
mplsp	*Cmd0051*	*optional*	*Logging SMS period*
mplsst	*Cmd0052*	*optional*	*Logging SMS starting date & time*
mplset	*Cmd0053*	*optional*	*Logging SMS ending date & time*
mpdl	*Cmd061*	*optional*	*Debugging level*
mpsh	*cmd0062*	*optional*	*Sms header*
mpfm	*cmd0063*	*optional*	*Functioning mode*
mpp1	*cmd0064*	*optional*	*Price 1 setting*
mpp2	*cmd0065*	*optional*	*Price 2 setting*
mpp3	*cmd0066*	*optional*	*Price 3 setting*
mpp4	*cmd0067*	*optional*	*Price 4 setting*
mppi1	*cmd0069*	*optional*	*Interim Price 1 setting*
mppi2	*cmd0070*	*optional*	*Interim Price 2 setting*
mppi3	*cmd0071*	*optional*	*Interim Price 3 setting*
mppi4	*cmd0072*	*optional*	*Interim Price 4 setting*
mpw	*Cmd0079*	*optional*	*Wear counter*

c. Statistics

Command	Alias	Parameters	Description
msf1	Cmd0080	optional	Global counter for A4 BW (format 1)
msf2	Cmd0081	optional	Global counter for A3 BW (format 2)
msf4	Cmd0082	optional	Global counter for A4 C (format 3)
msf4	Cmd0083	optional	Global counter for A3 C (format 4)
msfall	Cmd0084	optional	Prints the 4 formats
msif1	Cmd0085	optional	Global counter for A4 BW (format 1)
msif2	Cmd0086	optional	Global counter for A3 BW (format 2)
msif4	Cmd0087	optional	Global counter for A4 C (format 3)
msif4	Cmd0088	optional	Global counter for A3 C (format 4)
msifall	Cmd0089	optional	Prints the 4 formats
mswf1	Cmd0090	optional	Global counter for A4 BW (format 1)
mswf2	Cmd0091	optional	Global counter for A3 BW (format 2)
mswf4	Cmd0092	optional	Global counter for A4 C (format 3)
mswf4	Cmd0093	optional	Global counter for A3 C (format 4)
mswfall	Cmd0094	optional	Prints the 4 formats
msc	cmd0095	optional	Get or set credit
mst	cmd0096	optional	Get or set turnover
msw	cmd0099	optional	wear

d. Counters

Command	Alias	Parameters	Description
mchr	Cmd0100	optional	Hardware resets
mcsr	Cmd0101	optional	Software resets
mcee	Cmd0102	optional	Memory errors
mcsmss	Cmd0103	optional	SMS sent
mcsmsss	Cmd0104	optional	SMS sent with success
mcsmssf	Cmd0105	optional	SMS sent with failure
mcsmsr	Cmd0106	optional	SMS received
mcpont	Cmd0107	optional	Date & time of power on event
mcpofft	Cmd0108	optional	Date & time of power off event
mclsmss	Cmd0109	optional	Logged SMS
mclsmsss	Cmd0110	optional	Logged SMS with success
mclsmssf	Cmd0111	optional	Logged SMS with failure

mclsmslt	*Cmd0112*	*optional*	*Date & time of last logged SMS*
mclsmsnt	*Cmd0113*	*optional*	*Date & time of next SMS to be logged*
mcw	*Cmd0119*	*optional*	*Wear counter*

e. factory defaults

Command	*Alias*	*Parameters*	*Description*
mfdp	*Cmd0030*	*No*	*Factory default parameters*
mfds	*Cmd0031*	*No*	*Factory default statistics*
mfdc	*Cmd0032*	*No*	*Factory default counters*
mfdm	*Cmd0033*	*No*	*Factory default modem parameters*
mfdish	*Cmd0034*	*No*	*Factory default incoming SMS*
mfdosh	*Cmd0035*	*No*	*Factory default outgoing SMS*
mfdich	*Cmd0036*	*No*	*Factory default incoming CALLS*
mfdoch	*Cmd0037*	*No*	*Factory default outgoing CALLS*
mfdall	*Cmd0037*	*no*	*Factory default ALL*

3. **Wireless**

a) modem parameters

Command	*Alias*	*Parameters*	*Description*
mmpin	*Cmd0054*	*optional*	*Sim pin code*
mmpuk	*Cmd0055*	*optional*	*Sim puk code*
mmsp	*Cmd0056*	*optional*	*Subscriber phone*
mmmcp1	*Cmd0057*	*optional*	*Monitoring center phone1*
mmmcp2	*Cmd0058*	*optional*	*Monitoring center phone2*
mmscp1	*Cmd0059*	*optional*	*servicing center phone1*
mmscp2	*Cmd0060*	*optional*	*servicing center phone2*

b) SMS

Command	*Alias*	*Parameters*	*Description*
smssm	*Cmd0120*	*Mandatory*	*Send message*
smscm	*Cmd0121*	*optional*	*Count messages*
smsrm	*Cmd0122*	*Mandatory*	*Read message*
smsrmb	*Cmd0123*	*Mandatory*	*Read message (brute format)*
Smsrmall	*Cmd0124*	*optional*	*Read all messages*
smsrmball	*Cmd0125*	*optional*	*Read all messages (brute format)*
smsdm	*Cmd0126*	*Mandatory*	*Delete message*

smsdmall	*Cmd0127*	*optional*	*Delete all messages*
smsgl	*cmd0128*	*no*	*Get log*
smsglrf	*cmd0129*	*Mandatory*	*Get log using rf module*
smsglall	*cmd0130*	*no*	*Get all logs*
smssl	*cmd0131*	*optional*	*Sms send log*
smsglrf	*cmd0132*	*Mandatory*	*Sms send log rf*
smsslall	*cmd0133*	*optional*	*Sms send all logs*

c) Calls

Command	*Alias*	*Parameters*	*Description*
mdial	*Cmd0140*	*Mandatory*	*Make a phone call*
mhang	*Cmd0141*	*No*	*Hang the call*
manswer	*Cmd0142*	*No*	*Answer a call*

d) modem info

Command	*Alias*	*Parameters*	*Description*
mssca	*Cmd0150*	*optional*	*SMS service center*
msni	*Cmd0151*	*No*	*Modem serial number identification*
mcsq	*Cmd0152*	*No*	*Signal strength*
mtemp	*Cmd0153*	*No*	*Modem temperature*
mbatt	*Cmd0154*	*No*	*Modem battery level*

e) radio frequency

Command	*Alias*	*Parameters*	*Description*
Rfsch	*Cmd0160*	*No*	*Search for a wireless channel*
Rfconfig	*Cmd0161*	*No*	*Print configuration*
Rftest	*Cmd0162*	*No*	*Start Rf testing*

4. **Crypto**

Command	*Alias*	*Parameters*	*Description*
crptc	*Cmd0170*	*No*	*Crypto sanity check*
crpts	*Cmd0171*	*No*	*Serial number*
crptid	*Cmd0172*	*No*	*identity*

5. **Input/output**

Command	*Alias*	*Parameters*	*Description*
iorly1	*Cmd0180*	*Optional*	*Change relay1 status*
iorly2	*Cmd0181*	*Optional*	*Change relay2 status*

Ioop	*Cmd0182*	*No*	*read optos state*
Iopls	*Cmd0183*	*No*	*Read opto pulse state*
Iofmt	*Cmd0184*	*No*	*Read opto format state*
Iocol	*Cmd0185*	*No*	*Read opto color state*
Iomrst	*Cmd0186*	*Optional*	*Change modem reset signal state*
Iopk	*Cmd0187*	*Optional*	*Change modem pwrkey signal state*
iomdtr	*Cmd0188*	*Optional*	*Change modem dtr signal state*

6. **History**

Command	*Alias*	*Parameters*	*Description*
Mhic	*Cmd0190*	*No*	*List incoming calls*
Mhoc	*Cmd0191*	*No*	*List outgoing calls*
Mhis	*Cmd0192*	*No*	*List incoming SMS*
Mhos	*Cmd0193*	*No*	*List outgoing SMS*

7. **General**

Command	*Alias*	*Parameters*	*Description*
Acs	*Cmd0200*	*No*	*Check system settings*
Aip	*Cmd0201*	*No*	*Info. Parameters*
Aic	*Cmd0202*	*No*	*Info. Counters*

g

8. **Debug**

a. Trace

Command	*Alias*	*Parameters*	*Description*
shm	*Cmd0900*	*No*	*Shell mode*
tdpf	*Cmd0901*	*No*	*Trace default procedures*
tef	*Cmd0902*	*No*	*Trace events*
titf	*Cmd0903*	*No*	*Trace idle tasks*

b. Debug

Command	*Alias*	*Parameters*	*Description*
mdic	*Cmd1030*	*No*	*dump incoming calls*
mdoc	*Cmd1031*	*No*	*dump outgoing calls*
mdis	*Cmd1032*	*No*	*dump incoming sms*
mdos	*Cmd1033*	*No*	*dump outgoing sms*