**Testing :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| AT | Checking communication between the module and computer. |

**Call control :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| ATA | Answer command |
| ATD | Dial command |
| ATH | Hang up call |
| ATL | Monitor speaker loudness |
| ATM | Monitor speaker mode |
| ATO | Go on-line |
| ATP | Set pulse dial as default |
| ATT | Set tone dial as default |
| AT+CSTA | Select type of address |
| AT+CRC | Cellular result codes |

**Data card Control :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| ATI | Identification |
| ATS | Select an S-register |
| ATZ | Recall stored profile |
| AT&F | Restore factory settings |
| AT&V | View active configuration |
| AT&W | Store parameters in given profile |
| AT&Y | Select Set as power up option |
| AT+CLCK | Facility lock command |
| AT+COLP | Connected line identification presentation |
| AT+GCAP | Request complete capabilities list |
| AT+GMI | Request manufacturer identification |
| AT+GMM | Request model identification |
| AT+GMR | Request revision identification |
| AT+GSN | Request product serial number identification (IMEI) |

**Phone control :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| AT+CBC | Battery charge |
| AT+CGMI | Request manufacturer identification |
| AT+CGMM | Request model identification |
| AT+CGMR | Request revision identification |
| AT+CGSN | Request product serial number identification |
| AT+CMEE | Report mobile equipment error |
| AT+CPAS | Phone activity status |
| AT+CPBF | Find phone book entries |
| AT+CPBR | Read phone book entry |
| AT+CPBS | Select phone book memory storage |
| AT+CPBW | Write phone book entry |
| AT+CSCS | Select TE character set |
| AT+CSQ | Signal quality |

**Computer data interface :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| ATE | Command Echo |
| ATQ | Result code suppression |
| ATV | Define response format |
| ATX | Response range selection |
| AT&C | Define DCD usage |
| AT&D | Define DTR usage |
| AT&K | Select flow control |
| AT&Q | Define communications mode option |
| AT&S | Define DSR option |
| AT+ICF | DTE-DCE character framing |
| AT+IFC | DTE-DCE Local flow control |
| AT+IPR | Fixed DTE rate |

**Service :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| AT+CLIP | Calling line identification presentation |
| AT+CR | Service reporting control |
| AT+DR | Data compression reporting |
| AT+ILRR | DTE-DCE local rate reporting |

**Network Communication parameter :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| ATB | Communications standard option |
| AT+CBST | Select bearer service type |
| AT+CEER | Extended error report |
| AT+CRLP | Radio link protocol |
| AT+DS | Data compression |

**Miscellaneous :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| A/ | Re-execute command line |
| AT? | Command help |
| AT\*C | Start SMS interpreter |
| AT\*T | Enter SMS block mode protocol |
| AT\*V | Activate V.25bis mode |
| AT\*NOKIATEST | Test command |
| AT+CESP | Enter SMS block mode protocol |

**SMS Text mode :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| AT+CSMS | Select message service |
| AT+CPMS | Preferred message storage |
| AT+CMGF | Message format |
| AT+CSCA | Service centre address |
| AT+CSMP | Set text mode parameters |
| AT+CSDH | Show text mode parameters |
| AT+CSCB | Select cell broadcast message types |
| AT+CSAS | Save settings |
| AT+CRES | Restore settings |
| AT+CNMI | New message indications to TE |
| AT+CMGL  +CMGL[=*message\_status*]  **REC UNREAD**. It refers to the message status "received unread". It is the default value  **REC READ**. It refers to the message status "received read | List message  **STO UNSENT**  **STO SENT**  **ALL** |
| AT+CMGR | Read message |
| AT+CMGS | Send message |
| AT+CMSS | Send message from storage |
| AT+CMGW | Write message to memory |
| AT+CMGD | Delete message |

**SMS PDU mode :**

|  |  |
| --- | --- |
| **Command** | **Description** |
| AT+CMGL | List Messages |
| AT+CMGR | Read message |
| AT+CMGS | Send message |
| AT+CMGW | Write message to memory |

Examples of GSM commands:[[6]](http://en.wikipedia.org/wiki/Hayes_command_set#cite_note-sonyericsson_65054-6)[[7]](http://en.wikipedia.org/wiki/Hayes_command_set#cite_note-m2m-platforms_com_UC864E_guide-7)

|  |  |  |
| --- | --- | --- |
| **Command** | | **Description** |
| AT+CPIN=1234 | | Enter [PIN code](http://en.wikipedia.org/wiki/Personal_identification_number) |
| AT+CPWD="SC","old","new" | | Change PIN code from 'old' to 'new' |
| AT+CLCK="SC",0,"1234" | | Remove PIN code |
| AT&V | | Status |
| ATI | | Status (Manufacturer, Model, Revision, [IMEI](http://en.wikipedia.org/wiki/International_Mobile_Equipment_Identity), capabilities) |
| AT+COPS=? | | List available networks 0-Unknown/2-Current/3-Forbidden, Longname, Shortname, Numerical-ID, "AcT" |
| AT+CSQ | | Get signal strength. Answer: +CSQ: <[rssi](http://en.wikipedia.org/wiki/Received_signal_strength_indication) (more=better)>, <[ber](http://en.wikipedia.org/wiki/Bit_error_ratio), less=better> |
| ATD\*99# | | Dial access point |
| AT+CGDCONT=1,"IP","access.point.name" | | Defines [PDP context](http://en.wikipedia.org/wiki/GPRS_Core_Network#PDP_context). [[6]](http://en.wikipedia.org/wiki/Hayes_command_set#cite_note-sonyericsson_65054-6) |
| ATA | Answering in Voice Mode |
| ATD | Dial command in Voice Mode |
| ATH | Hang up in Voice Mode |
| ATZ | Reset from Voice Mode |
| AT#BDR | Select baud rate (turn off autobaud) |
| AT#CID | Enable Caller ID detection and select reporting format |
| AT#CLS | Select data, fax, or voice |
| AT#MDL? | Identify model |
| AT#MFR? | Identify manufacturer |
| AT#TL | Transmit level control |
| AT#REV? | Identify revision level |
| AT#RG | Record gain control |
| AT#SPK | Change the setting of Speakerphone |
| AT#VBS | Bits per sample (ADPCM) |
| AT#VBT | Beep tone timer |
| AT#VLS | Voice line select (ADPCM) |
| AT#VRA | Ringback goes away timer (originate) |
| AT#VRX | Voice Receive Mode (ADPCM) |
| AT#VSD | Silence deletion tuner (voice receive, ADPCM) |
| AT#VSP | Silence detection period (voice receive, ADPCM) |
| AT#VSS | Silence sensitivity tuner (voice receive) |
| AT#VTX | Voice Transmit Mode (ADPCM) |
| AT#VBQ? | Query buffer size |
| AT#VCI? | Identify compression method (ADPCM) |
| AT#VRN | Ringback never came timer (originate) |
| AT#VSK | Buffer skid setting |
| AT#VSR | Sampling rate selection (ADPCM) |
| AT#VTD | DTMF/tone reporting capability |
| AT#VTS | Play tone string (online voice command) |

Complements of the MagicSurfer 56k User's Guide. I/OMAGIC Corporation, 6B Autry, Irvine, CA 92618

**1.1. The 1st Way: Using a Computer to Receive SMS Messages through a Mobile Phone or GSM/GPRS Modem**

Receiving SMS messages through a mobile phone or GSM/GPRS modem has a major advantage over the other two ways -- wireless carriers usually do not charge any fees for receiving incoming SMS messages with their SIM cards. The disadvantage of receiving SMS messages this way is that a mobile phone or GSM/GPRS modem cannot handle a large amount of SMS traffic. One way to overcome this is to load balance the SMS traffic with a pool of mobile phones or GSM/GPRS modems. Each mobile phone or GSM/GPRS modem will have its own SIM card and mobile phone number.

In terms of programming, sending and receiving SMS messages through a mobile phone or GSM/GPRS modem are similar. What you need to do is to send instructions (in the form of AT commands) to the mobile phone or GSM/GPRS modem.

The following table lists the AT commands that are related to the receiving and reading of SMS messages:

| **AT command** | **Meaning** |
| --- | --- |
| +CNMI | New message indications |
| +CMGL | List messages |
| +CMGR | Read messages |
| +CNMA | New message acknowledgement |

Below shows a simple example that demonstrates how to use AT commands and the HyperTerminal program of Microsoft Windows to read SMS text messages received by a GSM / GPRS modem or mobile phone. The lines in bold type are the command lines that should be entered in HyperTerminal. The other lines are responses returned from the GSM / GPRS modem or mobile phone.

**AT**  
OK  
**AT+CMGF=1**  
OK  
**AT+CMGL="ALL"**  
+CMGL: 1,"REC READ","+85291234567",,"06/11/11,00:30:29+32"  
Hello, welcome to our SMS tutorial.  
+CMGL: 2,"REC READ","+85291234567",,"06/11/11,00:32:20+32"  
A simple demo of SMS text messaging.  
  
OK

Here is a description of what is done in the above example:

* Line 1: "AT" is sent to the GSM / GPRS modem to test the connection. The GSM / GPRS modem sends back the result code "OK" (line 2), which means the connection between the HyperTerminal program and the GSM / GPRS modem works fine.
* Line 3: The AT command +CMGF is used to instruct the GSM / GPRS modem to operate in SMS text mode. The result code "OK" is returned (line 4), which indicates the command line "AT+CMGF=1" has been executed successfully. If the result code "ERROR" is returned, it is likely that the GSM / GPRS modem does not support the SMS text mode. To confirm, type "AT+CMGF=?" in the HyperTerminal program. If the response is "+CMGF: (0,1)" (0=PDU mode and 1=text mode), then SMS text mode is supported. If the response is "+CMGF: (0)", then SMS text mode is not supported.
* Line 5-9: The AT command +CMGL is used to list all SMS text messages in the message storage of the GSM / GPRS modem. There are two SMS text messages in the message storage: "Hello, welcome to our SMS tutorial." and "A simple demo of SMS text messaging.". "+85291234567" is the sender mobile phone number. "06/11/11,00:30:29+32" and "06/11/11,00:32:20+32" tell us when the SMS text messages were received by the SMSC. "+32" is the time zone. Note that the unit is a quarter of an hour. So, +32 means GMT+8 hours, since 32 quarters of an hour = 8 hours. "REC READ" indicates both of the SMS text messages have been read before.
* Line 11: The result code "OK" indicates the execution of the AT command +CMGL is successful.

To enable an application to receive SMS messages, you have to write the source code for connecting to and sending AT commands to the mobile phone or GSM/GPRS modem, just like what a terminal program (such as HyperTerminal of Microsoft Windows) does. You can write the source code in C, C++, Java, Visual Basic, Delphi or other programming languages you like.

However, like what we have discussed in the earlier section "[The 1st Way: Sending SMS Messages from a Computer Using a Mobile Phone or GSM/GPRS Modem](http://www.developershome.com/sms/howToSendSMSFromPC.asp#10.1.The 1st Way: Sending SMS Messages from a Computer Using a Mobile Phone or GSM/GPRS Modem|outline)", usually a better solution is to use a high-level SMS messaging API (Application programming interface) / SDK (Software development kit) / library instead of writing your own code for interacting with the mobile phone or GSM/GPRS modem via AT commands. The API / SDK / library encapsulates the low-level details. So, an SMS application developer does not need to know AT commands and the composition of SMS messages in the bit-level. Some SMS messaging APIs / SDKs / libraries support SMSC protocols in addition to AT commands. To move from a wireless-modem-based SMS solution to a SMSC-based SMS solution, usually you just need to modify a configuration file / property file or make a few changes to your SMS messaging application's source code. The links to some open source and free SMS messaging libraries can be found in the article "[Free Libraries/Tools for Sending/Receiving SMS with a Computer](http://www.developershome.com/sms/freeLibForSMS.asp)".

Another high-level solution is to place an SMS gateway between the SMS messaging application and the mobile phone or GSM/GPRS modem. The SMS messaging application can then use simple protocols such as HTTP / HTTPS for receiving SMS messages. If an SMSC protocol (e.g. SMPP, CIMD, etc) is used for communicating with the SMS gateway instead of HTTP / HTTPS, an SMS messaging API / SDK / library can be very helpful to you since it encapsulates the SMSC protocol's details.

**AT+CPMS=?**

The modem will respond with a list of supported storages, for instance:

**+CPMS: (("SM","BM","SR"),("SM"))**

The storage can be selected using the following command:

**AT+CPMS=<read>[,<send>,<receive>] <ENTER>**

The first parameter sets the storage to read from the second optional specifies the storage to send messages from and the last optional parameter tells the device where to store newly received messages.

For example, to read messages from the SIM card use:

**AT+CPMS="SM" <ENTER>**

The modem should respond with the following string:

**+CPMS: <used\_space>,<max\_space> <ENTER>**

The used\_space indicates the number of messages currently in this memory, the max\_space the number of messages that can be stored.